COMMODORE

MENSILE PER UTENTI DI VIC

Lire 3000

TELEFONO

IL PROBLEMA DEL PENDOLO

MICROSCOPIO

STATISTICA

Febbraio/Marzo 1985 - Anno III N°7 - Sped. Abb. Post. Gr. III/70 (CR) - Distr. MePe

COMMODORE

DIRETTORE
Gloriano Rossi

REDAZIONE/COLLABORATORI
Eugenio Coppari, Giancarlo De Cobelli, Marco De Martino, Marco De Rosa, Valerio Ferri, Francesco Gatti, Mariangela Guardione, Giulio Marcozzi, Mauro Massetti, Ernesto Sidoti, Renzo Zonin.

SEGRETERIA DI REDAZIONE
Maura Ceccaroli, Piera Perin

GRAFICA e IMPAGINAZIONE
Francesco Amatori, Renato Caruso

EDIZIONI
Systems Editoriale S.r.l.

DIFFUSIONE e ABBONAMENTI
Marina Vantini

DIREZIONE, REDAZIONE, PUBBLICITA'
Viale Famagosta, 75 - 20142 Milano
Tel. 02/8467348 - Autorizzazione del Tribunale di Milano N. 103 del 25/2/84
Direttore responsabile:
Agostina Ronchetti

PUBBLICITA'
• Milano: Mirco Croce (coordinatore), Giuseppe Porzani, Michela Prandini, Giorgio Ruffoni, Claudio Tidone, Villa Claudio
Segretaria: Liliana De Giorgio
• Roma: Spazionuovo
Via P. Foscari, 70 - Tel. 06/8109679

STAMPA
Litografica - Busto Arsizio

Concessionario esclusivo per la diffusione MEPE Spa Via G. Carcano, 32 Milano

Spedizione in abbonamento postale Gruppo III/70

Prezzo della rivista L. 3.000
Numero arretrato L. 6.000
Abbonamento annuo L. 28.000
I versamenti vanni indirizzati a: Commodore C.C. V.le Famagosta, 75 20142 Milano, mediante assegno bancario, utilizzando il c/c postale N. 31532203

Per i cambi di indirizzo, indicare, oltre naturalmente il nuovo, anche l'indirizzo precedente, ed allegare alla comunicazione l'importo di L. 500 anche in francobolli.

LA POSTA

● Posseggo un registratore SIRIUS che purtroppo non funziona molto bene. Infatti, dopo aver fatto allineare le testine da personale specializzato per ben tre volte, in fase di caricamento, genera un ?LOAD ERROR. Premetto che subito dopo l'allineamento funziona a dovere per alcuni giorni (perciò il mio C64 funziona a dovere). Cosa posso fare?

(Ermanno Salvalaio)

☐ Tante volte non è colpa del registratore, ma delle cassette.
Tante volte non è colpa delle cassette, ma del registratore.
E allora?
Un sistema pratico per individuare il colpevole è quello di farsi imprestare un registratore originale Commodore di sicuro funzionanto.
Se il registratore SIRIUS risultasse colpevole allora la mia affermazione che conferma l'opportunità di utilizzo di optionals originali si fa sempre più forte. Molto spesso infatti per rispamiare qualche lira si rischiano inconvenienti di questo genere.
Se l'accusa si focalizza sulla cassetta allora è necessario riregistrarla più correttamente o farsela cambiare con un'altra di sicuro caricamento.

● Ho preso il PIN—1 (massa) del connettore del tape ed il PIN—4(lettura cassetta) ed ho fatto uno spinotto l'ho collegato ad un registratore mono, il tutto senza alcuna interfaccia, eppure ho comodamente duplicato tante cassette (anche Commodore Club N.3) che hanno il volume troppo basso. E' logico tutto ciò? O è solo un caso?

(Claudio Veneziani)

☐ No non è logico! Tutto dipende dalla qualità del tape che probabilmente non è alta. Un registratore di "corsa" potrebbe registrare un segnale con una tale distorsione che potrebbe essere ben digerita dal Computer. Qui si dovrebbe eseguire una lunga dissertazione sulla distorsione armonica, sull'onda quadra, etc, ma senza alcun dubbio sarebbe tediosa.

● Sono un felice possessore di VIC-20. Ho intenzione di comprare una penna ottica, ma a che cosa può servire? Gradirei informazioni. Grazie.

(Nino Tricoli)

☐ Una penna ottica può essere utile in molteplici applicazioni. Tutto dipende dal software che si fa girare. Disegnare, leggere i codici a barre, rilevare delle coordinate e tante altre applicazioni.

● E' possibile collegare insieme il VIC-20 ed il Commodore 64? Si potrebbe per esempio continuare a programmare con il 64 mentre il VIC stampa o carica un'altro programma.

(Mario Comuzzo)

☐ Sì, ma il discorso è alquanto complesso.
Per dare una risposta più precisa occorre anche sapere quali possano essere le periferiche in comune.
In ogni caso l'accesso alla stampante o al floppy ed ancora al plotter è possibile solo uno alla volta. Quando accede uno dei computer non è possibile, per l'altro, fare altrettanto.

● Vorrei sapere se il Sofware del Commodore 64 è compattibile con il Commodore 16 e se no, vi prego di indicarmi quale è disponibile sul mercato. Vorrei anche sapere come avviene il collegamento tramite Modem fra computer e se il Commodore 16 ed il VIC—20 possono comunicare con altri sia piccoli, medi o grandi sistemi.

(Mauro Imbriani)

☐ La compattibilità fra il C16 ed il C64 può essere considerata totale se le istruzioni BASIC utilizzate sul C64 non prevedono le PEEK, le POKE e le SYS.
Infatti l'interprete del C16 è una versione potenziata del BASIC del 64.
Quali allora le differenze? Molte!
— C16 maggiore facilità di utilizzo della grafica
— C64 migliore qualità del "sound"
— C16 inesistenza degli SPRITE
— C64 mancanza del Monitor in L.M.
etc.etc.etc.
Per ciò che riguarda il modem tutto dipende dal sofware di comunicazione. Si possono trasferire dati oppure programmi oppure ancora dare dei comandi.

● Ho acquistato la cassetta che fa parlare il mio Commodore 64, ma non riesco a far partire il programma anzi il mio C64 non segnala nemmeno l'apertura dei files. Quindi chiedo come devo fare a far partire i programmi. Grazie.

(Dante Di Senno)

☐ Il programma "La Voce" è un prodotto di una grande complessità. E' stato scritto interamente in linguaggio macchina (Assembler). Tranne la parte di caricamento e pilotaggio.
Ecco perchè è tassativo caricare LA VOCE solo dopo aver spento e riacceso il C64 in questa maniera:
LOAD"LA VOCE"quindi RETURN
ed in seguito quando riappare il cursore

occorre digitare:
RUN seguito dal tasto RETURN
Accadrà ora il caricamento automatico della parte in L.M. (programma ROB). Ciò richiederà circa 5 minuti.

● Dove posso trovare il "C16 Programmer's Guide"? Il manuale in dotazione alla macchina è quasi inutile!

(Michele Reboli)

☐ Il manuale in questione non è ancora disponibile sul mercato italiano. Occorre sperare che tale realizzazione libraria possa divenire realtà al più presto.

● Chi mi può aiutare in GOLDEN BATON? Sono nella caverna!

(Pendini Marco)

☐ Lancia un grido di disperazione!
Bando agli scherzi. Non lo so caro Marco.
Però quale soddisfazione ne trarresti se, avendo la soluzione, ti svelassi l'arcano?

● Vorrei sapere se è possibile far funzionare un VIC-20 senza modulatore R.F. e, siccome io ne sono sprovvisto, vorrei sapere dove posso acquistarlo visto che da queste parti non si trova.

(Domenico Principe)

☐ Rispondendo a questa domanda ritengo di poter chiarire alcuni dubbi inerenti questo argomento. Sia il VIC 20 che il C64, nonché il C16 ed il PLUS4 posseggono sul retro della tastiera alcuni connettori che ci permettono di avere funzioni ben specifiche. Uno di questi connettori, come è spiegato nei relativi manuali, offre la possibilità di collegare direttamente un Monitor/tv.
Solo sul VIC-20 esiste la necessità di utilizzo un modulatore a radio frequenza, che negli altri modelli è presente all'interno della tastiera. In mancanza di questo "aggeggio" si può adattare un monitor oppure si può acquistare un modulatore R.F., anche di altro tipo, ed adattarlo o farlo adattare al VIC-20.

● Possiedo un Commodore 64 da circa 6 mesi e negli ultimi tempi succede che quando uso il tasto RETURN compaiono delle freccette rendendo impossibile il funzionamento, quando schiaccio i tasti per lo spostamento del cursore su e giù appare la scritta LOAD PRESS PLAY ON TAPE. Cosa devo fare?

(Cravanzola Cristiano)

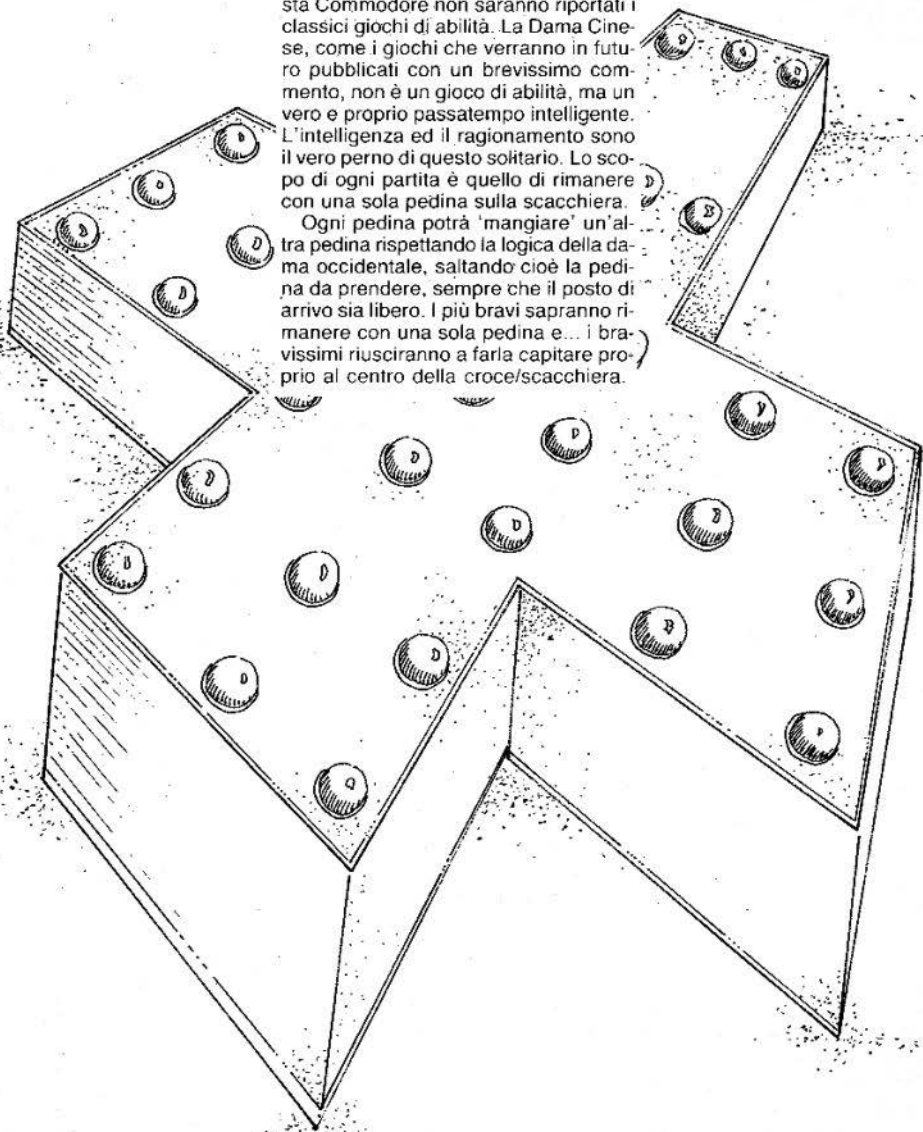
☐ Questo tipo di inconveniente può essere causato sia dall'uso prolungato della tastiera in maniera piuttosto "pesante" (famoso è il termine "digitazione da elefante o ippopotamo"). Un'altra causa potrebbe essere individuata da una alterazione dell'integrato decodificatore di tasto premuto, oppure dal CIA (6526).
Il suo caso specifico però mi lascia intuire che il guasto possa essere identificato nella seconda ipotesi e quindi è necessario che il suo C64 necessiti di un centro di assistenza Hardware per i Commodore. Ultima ipotesi: alcuni joystick, con ripetizione automatica di sparo inserita e collegati alla porta 1, provocano scherzi analoghi. Togliere quindi la ripetizione od il joy.

LA POSTA

DAMA CINESE

di **Gloriano Rossi (i2KH)**

Come più volte affermato, sulla rivista Commodore non saranno riportati i classici giochi di abilità. La Dama Cinese, come i giochi che verranno in futuro pubblicati con un brevissimo commento, non è un gioco di abilità, ma un vero e proprio passatempo intelligente. L'intelligenza ed il ragionamento sono il vero perno di questo solitario. Lo scopo di ogni partita è quello di rimanere con una sola pedina sulla scacchiera.

Ogni pedina potrà 'mangiare' un'altra pedina rispettando la logica della dama occidentale, saltando cioè la pedina da prendere, sempre che il posto di arrivo sia libero. I più bravi sapranno rimanere con una sola pedina e... i bravissimi riusciranno a farla capitare proprio al centro della croce/scacchiera.

```
100 REM ***********************
105 REM *                     *
110 REM * IL GIOCO DELLA      *
115 REM *                     *
120 REM *    D A M A          *
125 REM *                     *
130 REM *        C I N E S E  *
135 REM *                     *
140 REM ***********************
145 REM * AUTORE :  I2KH      *
150 REM *    GLORIANO ROSSI   *
155 REM ***********************
160 REM * VIC20            SI *
165 REM * COMMODORE 64     SI *
170 REM * C16              SI *
175 REM * PLUS 4           SI *
180 REM * SERIE 4000/8000 SI *
185 REM ***********************
190 DIM B(70),T(9,9),A$(14)
195 SP$="  ":REM  PER VIC20 =" "
200 SP=LEN(SP$)
205 A$=SP$+" ●":B$=SP$+" O"
210 A$(00)=SP$+SP$+SP$+SP$:IF LEN(S
    P$)=1 THEN  A$(00)=A$(00)+SP$+S
    P$
215 A$(02)=A$(00)+"13"+SP$+"14"+SP$
    +"15"
220 A$(04)=A$(00)+"22"+SP$+"23"+SP$
    +"24"
225 A$(06)="29"+SP$+"30"+SP$+"31"+S
    P$+"32"+SP$+"33"+SP$+"34"+SP$+"
    35"
230 A$(08)="38"+SP$+"39"+SP$+"40"+S
    P$+"41"+SP$+"42"+SP$+"43"+SP$+"
    44"
235 A$(10)="47"+SP$+"48"+SP$+"49"+S
    P$+"50"+SP$+"51"+SP$+"52"+SP$+"
    53"
240 A$(12)=A$(00)+"58"+SP$+"59"+SP$
    +"60"
245 A$(14)=A$(00)+"67"+SP$+"68"+SP$
    +"69"
250 PRINT"[CLEAR]"
255 FOR I=1 TO 14 STEP 1:PRINT TAB(
    10*(SP-1)+1)A$(I)
260 :IF I/2=INT(I/2) THEN PRINT
265 NEXTI
270 REM  *****************
275 REM  **    START    **
280 REM  *****************
285 FOR R=1 TO 9:FOR C=1 TO 9
290 :FOR C=1 TO 9
295 ::IF (R-4)*(R-5)*(R-6)=0 THEN 3
    15
300 ::IF (C-4)*(C-5)*(C-6)=0 THEN 3
    15
305 ::T(R,C)=-5
310 ::GOTO 325
315 ::IF (R-1)*(C-1)*(R-9)=0 THEN 3
    05
320 ::T(R,C)=5
325 :NEXTC
330 NEXTR
335 T(5,5)=0:GOSUB 445
340 FOR W=1 TO 33
345 :READ M
350 :DATA 13,14,15,22,23,24,29,30,3
    1,32,33,34,35,38,39,40,41
355 :DATA 42,43,44,47,48,49,50,51,5
    2,53,58,59,60,67,68,69
360 :B(M)=-7
365 NEXTW
370 B(41)=-3
375 PRINT"[HOME]
                   ";
380 INPUT "[HOME]DA";Z
385 PRINT"[HOME][3 DOWN]
                   ";
390 IF B(Z)=-7 THEN 400
395 PRINT"[HOME][3 DOWN]MOSSA ERRAT
    A":GOTO 375
400 INPUT "[HOME][3 DOWN] A";P
405 IF B(P)=0 THEN 395
410 IF B(P)=-7 THEN 395
415 IF Z=P THEN 375
420 IF ((Z+P)/2)=INT((Z+P)/2) THEN
    430
425 GOTO 395
430 IF (ABS(Z-P)-2)*(ABS(Z-P)-18)<>
    0 THEN 395
435 GOSUB 535:GOSUB 445:GOSUB 685:G
    OTO 375
440 REM  *****************
445 REM  * AGGIORNA LA    *
450 REM  * VISUALIZZAZIONE*
455 REM  *****************
460 PRINT"[HOME]"
465 FOR X=1 TO 9
470 :FOR Y=1 TO 9
475 ::IF (X-1)*(X-9)*(Y-1)*(Y-9)=0
    THEN 510
480 ::IF ((X-4)*(X-5)*(X-6)=0) OR (
    (Y-4)*(Y-5)*(Y-6)=0) THEN 490
485 ::GOTO 510
490 ::IF T(X,Y)<>5 THEN 505
495 ::PRINT TAB(8*(SP-1)+Y+NN);A$;
```

```
500 ::GOTO 510
505 ::PRINT TAB(8*(SP-1)+Y+NN),B$,
510 :NEXTY
515 :IF X>1 AND X<9 THEN PRINT"[2 D
    OWN]"
520 :NN=-2+(1*(SP-1)):IF X<3 OR X>5
     THEN NN=2*(SP)+NN+2
525 NEXTX:RETURN
530 REM  ******************
535 REM  * AGGIORNAMENTO   *
540 REM  * DELLE TABELLE   *
545 REM  ******************
550 C=1
555 FOR X=1 TO 9
560 :FOR Y=1 TO 9
565 ::IF C<>Z THEN 660
570 ::IF C+2<>P THEN 595
575 ::IF T(X,Y+1)=0 THEN 395
580 ::T(X,Y+2)=5
585 ::T(X,Y+1)=0:B(C+1)=-3
590 ::GOTO 650
595 ::IF C+18<>P THEN 615
600 ::IF T(X+1,Y)=0 THEN 395
605 ::T(X+2,Y)=5:T(X+1,Y)=0:B(C+9)=
    -3
610 ::GOTO 650
615 ::IF C-2<>P THEN 635
620 ::IF T(X,Y-1)=0 THEN 395
625 ::T(X,Y-2)=5:T(X,Y-1)=0:B(C-1)=
    -3
630 ::GOTO 650
635 ::IF C-18<>P THEN 660
640 ::IF T(X-1,Y)=0 THEN 395
645 ::T(X-2,Y)=5:T(X-1,Y)=0:B(C-9)=
    -3
650 ::B(Z)=-3:B(P)=-7
655 ::T(X,Y)=0:GOTO 675
660 ::C=C+1
665 :NEXTY
670 NEXTX
675 RETURN
680 REM  ******************
685 REM  * IL GIOCO E'     *
690 REM  ******************
695 REM  *    FINITO ?     *
700 REM  ******************
705 F=0
710 FOR R=2 TO 8
715 :FOR C=2 TO 8
720 ::IF T(R,C)<>5 THEN 810
725 ::F=F+1
730 ::FOR A=R-1 TO R+1
735 :::T=0
740 :::FOR B=C-1 TO C+1
745 ::::T=T+T(A,B)
750 :::NEXTB
755 :::IF T<>10 THEN 765
760 :::IF T(A,C)<>0 THEN 875
765 ::NEXTA
770 ::FOR X=C-1 TO C+1
775 :::T=0
780 :::FOR Y=R-1 TO R+1
785 ::::T=T+T(Y,X)
790 :::NEXTY
795 :::IF T<>10 THEN 805
800 :::IF T(R,X)<>0 THEN 875
805 ::NEXTX
810 :NEXTC
815 NEXTR
820 REM  ******************
825 REM  *      FINE       *
830 REM  ******************
835 PRINT"[CLEAR][3 DOWN]  IL  GIOC
    O  E'  FINITO"
840 PRINT"[3 DOWN]RIMANGONO ",F;" P
    EZZI"
845 IF F<>1 THEN 855
850 PRINT"[3 DOWN]ECCEZIONALE BRAVI
    SSIMO"
855 PRINT"[5 DOWN]VUOI GIOCARE ANCO
    RA ";:INPUT A$
860 IF LEFT$(A$,1)="N" THEN 880
865 RESTORE : RUN
870 STOP
875 RETURN
880 PRINT"[4 DOWN]   C I A O "
885 END
```

Cornucopia

a cura di **Gloriano Rossi**

Prosegue in questa quarta puntata di Cornucopia la raccolta di $nn tutta redatta da Voi lettori.

Ancora una volta alcune idee sono ottime, altre un po' meno, altre ancora un po' scarsine.

Tutto però può servire e l'idea di qualcuno può andare bene per un altro.

Continuate così!

La rubrica Cornucopia è Vostra in tutto e per tutto, con un'unica eccezione: il mio controllo di validità ed esattezza delle notizie.

Continuate ad inviare i vostri $nn a:

Spett. Rivista COMMODORE
rubrica Cornucopia
Gloriano Rossi
Viale Famagosta 75
20142 Milano

$30

PRINT USING? Questo breve programma serve per formattare numeri e stamparli allineati a destra. Il programa è fatto per la stampante VC-1525 ma può andar bene anche per qualsiasi altro tipo di output.

```
10 INPUT "[CLR][CSRS DOWN] NUMERO
   (MAX 9 CIFRE)";N:N$=STR$(N): L=LEN (N$)-1
20 IF L>=3 THEN 45
30 FOR I=3 TO L STEP 4:L$="."+RIGHT$ (N$,I)
40 N=INT (N/1000): N$=STR$(N):N$=N$+L$:NEXT
50 L=LEN (N$): P$=CHR$ (16)
60 C$=RIGHT$ (STR$(C),2):IFC (10 THEN
   C$="0"+ RIGHT$ (C$,1)
70 OPEN 4,4:PRINT =4,P$C$N$
```

(Filippo Tiberio)

$31

BLOCK FREE. Non so se può essere utile per qualcuno sapere che digitando LOAD"$ (ed una qualsiasi lettera dopo il dollaro)", 8 si hanno direttamente di blocchi liberi senza vedere la directory.

(ignoto)

$32

Lista variabili. Questo programma stampa il nome di tutte le variabili decimali usate da un programma in BASIC.
E' sufficiente inserire in coda al programma da testare questa piccola routine e dare il RUN (del numero di riga in questione).

(Gioiano Creatini)

```
1 A=PEEK(45)+PEEK(46)*256:B=PEEK(4
  7)+PEEK(48)*256:FOR X=A TO B STE
  P 7:PRINTCHR$(PEEK(X));CHR$(PEEK
  (X+1)):NEXT
```

$33

Bit per Bit. La cornucopia in questione serve a visualizzare il contenuto di un Byte facendolo vedere Bit per Bit.

(Luca Marcelli)

```
10 INPUT "BYTES";A$:IF A$="*" THEN
   END
20 A=VAL(A$):IF A<0  OR  A>65535 TH
   EN  10
30 B=PEEK(A)
40 FOR I=1 TO 8:A1=1
50 C=INT(B/2):IF B=C*2 THEN A1=0
60 PRINTA1;:B=C:NEXT:PRINT:GOTO 10
```

$34

Print AT. Tante volte è necessario stampare qualcosa in una precisa parte dello schermo ed allora si deve ricorrere ad una miriade di caratteri come quelli dei cursori tra virgolette, operazione alquanto noiosa. Questa semplice Cornucopia evita tutto ciò, basta dare le coordinate nella riga 20: X=riga Y=colonna.
Il valore di X non deve naturalmente superare 24 e quello di Y deve essere minore di 40.

(Enzo Tumino)

```
10 PRINTCHR$(147)
20 X=11:Y=14
30 IF X>24 OR Y>39 THEN PRINT"ERROR
   E NELLE COORDINATE":END
40 POKE 781,X:POKE 782,Y:POKE 780,0
   :SYS65520
50 PRINT"CORNUCOPIA"
```

$35

Questa cornucopia crea una **musichetta,** se si può chiamar così, generata casualmente dal computer.
Riga 20 e 30 si stabilisce il livello sonoro ed l'attack.
Riga 40 e 50 creano i valori casualmente.
Riga 60 HIGH BYTE,X: LOW BYTE,Y
Riga 70 forma d'onda ON
Riga 80 ciclo di durata nota.
Riga 90 forma d'onda OFF
Riga 99 ripetizione

(Simon Gadaleta)

```
20 POKE 54796,15
30 POKE 54277,9
40 X=INT(RND(1)*150)
50 Y=INT(RND(1)*150)
60 POKE 54273,X:POKE 54272,Y
70 POKE 54276,17
80 FOR I=1 TO 150:NEXT
90 POKE 54276,0
99 GOTO 40
```

$36

Ricopiando la ROM del BASIC nella sottostante RAM, è possibile tramite delle POKE modificare, o meglio personalizzare intere istruzioni BASIC. (questo concetto è già stato visto in maniera più concisa). Purtroppo le condizioni ritornano alla normalità premendo contemporaneamente RUN/STOP - RESTORE o digitando in maniera diretta POKE 1,55. Il ripristino alle condizioni normali può essere eliminato digitando POKE 808,251 (disabilitazione dei tasti RUN/STOP - RESTORE) e

19 - 23 APRILE 1985 - FIERA DI MILANO

Ogni anno circa due milioni di persone visitano la Fiera Campionaria di Milano. Dal 1985, ad attenderle, ci sarà COMPUTER SHOW il nuovo Salone interamente dedicato all'informatica per i giovani, la scuola, la famiglia moderna. Sicuramente sarà il più grande appuntamento dell'anno.

Perché non esserci?

la mostra per:
l'hobby
la didattica
i videogiochi

Segreteria:
E.P.I. - ENTE PROMOZIONE INFORMATICA.
Via Marochetti, 27 - 20139 Milano
Telefoni (02) 56.93.973 - 53.98.267

in seguito modificando il comando POKE agendo su opportune locazioni di memoria. Ecco come fare.

Il programma visualizza tutte le locazioni di memoria che vanno dall'indirizzo decimale 40960 in poi (indirizzo dell'area di sovrapposizione ROM-RAM) il relativo codice ASCII ed il carattere corrispondente al codice stesso:

```
10 FOR C=1 TO 25:PRINT"[DOWN]":NEXT
   C
20 A=40960
30 N=PEEK(A)
40 PRINTA,N
50 A=A+1
60 POKE 1970,N
70 GET R$:IF R$="" THEN 70
80 GOTO 30
```

Durante l'esecuzione del programma premete contemporaneamente SHIFT+COMMODORE (passaggio al minuscolo) e... tenendo premuta la barra spaziatrice vi renderete conto delle locazioni di memoria corrispondenti ai comandi END, FOR, NEXT, DATA allocate da 41118 in poi. Modificando, tramite delle POKE, i codici ASCII memorizzati nei comandi di LIST (loc. 41229), PEEK (loc. 41210), SAVE (loc. 41197), ed aggiungendolo, in cima al programma, renderete impossibile (tranne a coloro che conoscono le nuove parole chiavi) LISTare il programma, leggere le locazioni modificate (PEEK), riattivare la condizione normale, salvare il programma (SAVE) e chi più pazienza ha più ne metta. Per riassumere: le locazioni dei comandi da personalizzare vanno lette con il programma presentato e modificate con una POKE del tipo:
POKE X1,Z1 : POKE X2,Z2 : POKE Xn,Zn
dove X1, X2,...... Xn rappresentano le locazioni del comando da modificare (tante quanti i caratteri del comando) e Z1, Z2...... Zn i caratteri ASCII da modificare. Si supponga di voler modificare, per esempio, il comando LIST in CLIP. Si digiti dapprima NEW. E poi:

```
100 FOR N=40960 TO 49151
110 POKE N,PEEK(N):NEXT
120 POKE 1,54
130 POKE 41229,67:POKE 41230,76:POKE
     41231,73:POKE 41232,208
```

Le linee 100 e 110 ricopiano il BASIC nella memoria RAM. La linea 120 ci libera dalla ROM e la linea successiva modifica il nome del comando LIST in CLIP. E' di fondamentale importanza aggiungere il valore 128 all'ultimo codice da POKare nell'ultima locazione relativa al comando in oggetto.

Per cui nel caso di CLIP i codici ASCII da POKare nelle locazioni da 41229 a 41232 saranno rispettivamente 67 per C, 76 per L, 73 per I, 208 per P (80+128). Il programma seguente è l'esempio di come proteggo il mio software. Infatti questi permette...

La visualizzazione del comando RUN/STOP - RESTORE
La variazione del comando PEEK in LEGR
La variazione del comando LIST in VISL
La variazione del comando SAVE in TROV
La variazione del comando PRINT in STAMP
La variazione del comando CMD in USC
La variazione del comando OPEN in APRI
La variazione del comando CLOSE in CHIUD
La variazione del comando GOTO in RECT
La variazione del comando GOSUB in VARIT
La variazione del comando POKE in MODF

Ho ritenuto opportuno modificare, anche se non era del tutto necessario, anche i comandi GOTO e GOSUB, poichè sarebbe stato possibile lanciare il programma evitando le righe di modifica. Dopo aver lanciato il programma è necessario attendere circa un minuto finchè non compare la scritta READY. Provate a saggiare l'effetto delle nuove e vecchie istruzioni. Listando il programma (con il nuovo comando) dopo aver lanciato le modifiche almeno una volta, noterete dei cambiamenti nel listato. Niente paura. Basta digitare in modo diretto: X 808,237 Dove X rappresenta il comando che ha sostituito provvisoriamente l'istruzione POKE (nel mio caso MODF), dopo di che, premendo RUN/STOP - RESTORE e digitando LIST, lo rivedrete scorrere in tutto il suo splendore. Inoltre è consigliabile personalizzare anche l'istruzione RUN per gli stessi motivi per cui abbiamo modificato anche le istruzioni GOTO, GOSUB e quindi avvertire l'usufruista del programma di lanciarlo non più con il solito RUN bensì con la nuova istruzione. Oppure aggiungendo in coda al programma l'autoRUN.

(Ciro De Pasquale)

```
 10 FOR N=40960 TO 49151
 20 POKE N,PEEK(N):NEXT
 30 POKE 1,54
 40 POKE  808,251
 50 REM  * SAVE IN TROV *
 60 POKE 41197,83:POKE 41198,65:POKE
     41199,86:POKE 41200,197
 70 REM  * PRINT# IN STAMP# *
 80 POKE 41214,83:POKE 41215,84:POKE
     41216,65
 90 POKE 41217,77:POKE 41218,80:POKE
     41219,163
100 REM  * LIST IN VISL *
110 POKE 41229,86:POKE 41230,73:POKE
     41231,83:POKE 41232,204
120 REM  CMD IN USC *
130 POKE 41236,85:POKE 41237,83:POKE
     41238,195
```

```
140 REM  * OPEN IN APRI *
150 POKE 41242,65:POKE 41243,80:POKE
     41244,82:POKE 41245,201
160 REM  * CLOSE IN CHIUD *
170 POKE 41246,67:POKE 41247,72:POKE
     41248,73:POKE 41249,35:POKE 412
    50,196
180 REM  * PEEK IN LEGR *
190 POKE 41335,76:POKE 41336,69:POKE
     41337,71:POKE 41338,210
200 REM  * GOTO IN RECT *
210 POKE 41153,82:POKE 41154,69:POKE
     41155,67:POKE 41156,212
220 REM  * GOSUB IN VARIT *
230 POKE 41169,86:POKE 41170,65:POKE
     41171,82:POKE 41172,73:POKE 411
    73,212
240 REM  * POKE IN MODF *
250 POKE 41210,77:POKE 41211,79:POKE
     41212,68:POKE 41213,198
260 REM  INIZIO PROGRAMMA VERO E PRO
    PRIO
```

$37

Stop list con SIMON'S BASIC. Come tutti sappiamo il SIMON'S BASIC comprende (tra le altre 113 istruzioni) l'istruzione PAGE che consente di visualizzare un listato suddividendolo in pagine successive formate da un numero di righe impostato dall'utente. Talvolta però sarebbe comodo scorrere velocemente il listato per poi fermarlo in un punto desiderato. Oppure rallentare lo scroll in modo più evidente di quello che otteniamo comunemente premendo il tasto CTRL.

Ebbene in ambiente SIMON'S BASIC tutto ciò è possibile senza alcuna utility aggiuntiva. Carica un programma. Digita LIST e mentre le linee "scrollano" sul video premi SHIFT. Cosa è accaduto? Lo scroll è diventato lentissimo. Viene visualizzato un carattere alla volta o poco più.

Ma non è finita qui! Prova adesso a ripetere l'operazione, mantenendo premuto lo SHIFT (o per non stancare il tuo ditino SHIFT LOCK) premi CTRL e, come per magia il listato si fermerà e resterà lì immobile a tua disposizione fino a che non rilascerai il tasto SHIFT (o SHIFT LOCK).

(Maurizio Garutti)

$38

Le frazioni. E' nota l'impossibilità di assegnare alle variabili numeriche dei valori frazionari. Un modo ovvio per aggirare l'ostacolo è il seguente: inseriamo separatamente il numeratore ed il denominatore, il Computer eseguirà l'operazione.

Ciò comporta il fatto che siamo costretti ad inserire 1 come denominatore ogni volta che il numero non è frazionario, ma intero. Con la seguente routine è possibile inserire valori interi o frazionari. Con i frazionari il computer ci dà il corrispondente numero decimale. Il numeratore dovrà essere minore di 10.000, in caso contrario sono facilmente intuibili le righe da inserire.

(Salvatore Di Guida)

```
10 INPUT A$
20 IF LEN(STR$(VAL(A$)))>LEN(A$) TH
   EN PRINTVAL(A$):END
30 IF VAL(A$)<=9 THEN PRINTVAL(A$)/
   VAL(RIGHT$(A$,LEN(A$)-2)):END
40 IF VAL(A$)<=99 THEN PRINTVAL(A$)
   /VAL(RIGHT$(A$,LEN(A$)-3)):END
50 IF VAL(A$)<=999 THEN PRINTVAL(A$
   )/VAL(RIGHT$(A$,LEN(A$)-4)):END
60 IF VAL(A$)<=9999 THEN PRINTVAL(A
   $)/VAL(RIGHT$(A$,LEN(A$)-5)):END
```

$39

Colori in Assembler (1). Con questa routine gestirete i colori del bordo del Commodore 64 in maniera assai più veloce rispetto all'equivalente programma scritto in linguaggio BASIC. La maggiore rapidità è da addebitare al fatto che l'Assembler non necessita della fase di interpretazione tipica invece del BASIC. La velocità con cui mutano i colori sullo schermo non è completamente percepibile dall'occhio umano.

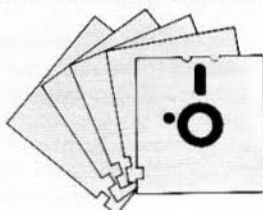
(Redazione)

```
10 POKE 4096,238:POKE 4097,32:POKE
   4098,208:POKE 4099,76:POKE 4100,
   0:POKE 4101,16:SYS4096
```

$3A

Colori in Assembler (2). Questo programma è simile al precedente. Unica differenza: viene fatto lampeggiare l'interno dello schermo del vostro Commodore 64.

(Redazione)

```
10 POKE 4096,238:POKE 4097,33:POKE
   4098,208:POKE 4099,76:POKE 4100,
   0:POKE 4101,16:SYS4096
```

IL COMMODORE 64 AL MICROSCOPIO

di **Marco De Rosa**

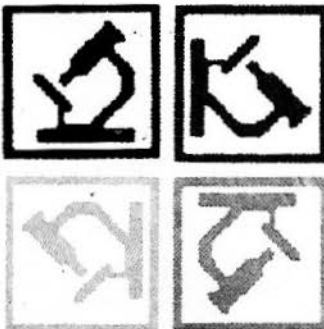
parte 3ª

(197) tasto corrente premuto

Questa locazione contiene il codice ASCII dell'ultimo tasto premuto e messo nel buffer di tastiera (vedi locazioni 198 e da 631 a 640). Quando viene vuotata, e nessun tasto è premuto, essa contiene il valore 64. Se vengono premuti più tasti contemporaneamente il valore memorizzato sarà quello del carattere con priorità più alta. Potete vedere in figura 1 l'elenco delle priorità.

(198) NUMERO DI CARATTERI PRESENTI NEL BUFFER DI TASTIERA

Questa locazione contiene il numero di caratteri presenti nel buffer di tastiera che si trova da 631 a 640. Il suo valore di default è zero, e cambia quando premete dei tasti e il computer sta eseguendo altre operazioni. In questo caso i codici ASCII dei tasti premuti vengono immagazzinati nelle locazioni da 631 a 640 e il numero di caratteri viene aumentato di uno. E' buona regola pulire questa locazione prima di ogni istruzione di GET con un POKE 198,0.

Un uso frequente di questa locazione viene fatto nei programmi di caricamento. Supponiamo di voler scrivere una routine che carichi un programma in linguaggio macchina a partire da 2048 ($0800) e lo faccia partire con una chiamata all'indirizzo giusto, diciamo 4096.

La prima cosa che viene in mente è scrivere:

```
10 LOAD"nome", 8,1
20 SYS 4096
```

Questo sistema purtroppo non può funzionare, perchè non appena viene eseguita l'istruzione di LOAD, il programma viene cancellato dal file caricato, e la chiamata alla riga 20 viene persa. Il sistema più usato è invece quello di scrivere le istruzioni di LOAD e SYS (ed altre eventuali) sullo schermo, il quale si trova in una zona della memoria (da 1024 a 2048) che non viene quasi mai usata per i programmi. Si forzano poi nel buffer di tastiera i vari RETURN (valore ASCII 13) necessari per eseguire queste istruzioni. Il numero di questi deve essere scritto nella locazione 198.

Il programma qui sotto carica, ad esempio, il file "nome" e lo fa partire con una SYS 4096.

```
20 PRINT"[CLR] LOAD"+CHR$(34)+
"nome"+CHR$(34)+",8,1"
30 PRINT"[5DOWN] SYS 4096"
40 POKE 198,3:POKE 631,19:
POKE 632,13:POKE 633,13
```

(199) FLAG CARATTERE IN REVERSE MODE

I due valori assunti da questa locasione sono:

0 per carattere normale

18 per carattere in reverse mode

E' più facile e più veloce scrivere POKE 199,1, o POKE 199,18 che usare i tasti RVS ON e RVS OFF.

(200-202) CURSORE IN OPERAZIONI DI INPUT

Queste tre locazioni contengono rispettivamente il numero della colonna schermo alla fine, della colonna e della riga all'inizio, di tutte le operazioni di INPUT. Purtroppo non è possibile POKE-are queste locazioni, perchè non hanno nessun effetto sulla posizione del cursore sullo schermo.

(204-207) LOCAZIONI CHE INTERESSANO IL CURSORE

Contengono rispettivamente:

204 - Lampeggio cursore: 0=acceso, 1=spento.

205 - Conto alla rovescia per lampeggio. Il suo valore di default è 2.

206 - Codice ASCII del carattere sotto il cursore. Il suo valore di default è 32.

207 - Ultimo lampeggio cursore. Il suo valore di default è 0.

Usando queste locazioni in modo opportuno è possibile costruire un cursore lampeggiante durante un'istruzione di GET.

```
10 PRINT"BATTI UN TASTO ";
20 GOSUB 100: GET A$:IF A$="" THEN 20
30 POKE 207,0:PRINT A$:GOTO 15
100 REM CURSORE
110 I=I+1
115 IF I=10 AND PEEK(204)=0 THEN POKE 204,1:I=0
120 IF I=10 AND PEEK(204)=1 THEN POKE 204,0:I=0
130 RETURN
```

(213) LUNGHEZZA DELLA LINEA FISICA DI SCHERMO

Questa locazione contiene il valore della lunghezza della linea fisica sullo schermo. Il valore di default è 39, ma è possibile POKE-are un numero minore ottenendo effetti interessanti. Provate ad esempio il seguente programma:

```
2 PRINT"[CLR]"
5 A$="01234567890123456789012345678901234567890"
10 FOR I=38 TO 20 STEP -1
20 PRINT A$
30 NEXT I
```

(da 217 a 242) LINK TABLE PER RIGHE DI SCHERMO

Come la maggior parte dei computers, il CBM 64 accetta righe BASIC più lunghe di una riga di schermo. Queste ultime possono essere al massimo di 40 caratteri e vengono chiamate "fisiche", le altre arrivano invece a lunghezza 80 e vengono dette 'logiche". Ogni qualvolta voi scrivete una riga logica più lunga di 40 caratteri, l'elaboratore memorizza in queste locazioni il link (collegamento) tra le due righe fisiche necessarie, e le tratta come se fossero una sola lunga riga.

N°	TASTO	N°	TASTO	N°	TASTO	N°	TASTO
0	INST	16	5	32	9	48	
1	RET	17	R	33	I	49	*
2	⇆CRSR	18	D	34	J	50	]
3	F 7	19	6	35	0	51	CLR
4	F 1	20	C	36	M	52	
5	F 3	21	F	37	K	53	=
6	F 5	22	T	38	O	54	↑
7	⇅ CRSR	23	X	39	N	55	/
8	3	24	7	40	+	56	1
9	W	25	Y	41	P	57	←
10	A	26	G	42	L	58	
11	4	27	8	43	-	59	2
12	Z	28	B	44	.	60	SPC
13	S	29	H	45	[	61	
14	E	30	U	46	@	62	Q
15		31	V	47	,	63	

Figura 1

(da 601 a 630) CANALI APERTI PER ROUTINES KERNAL

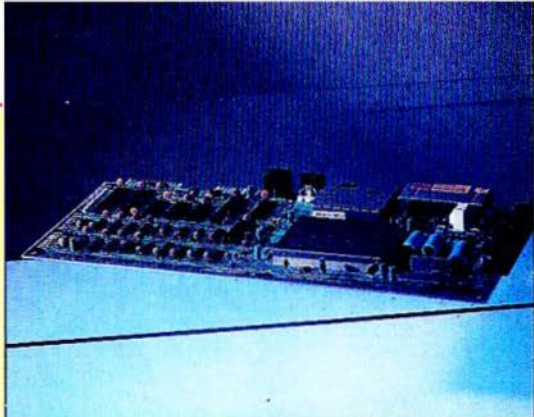
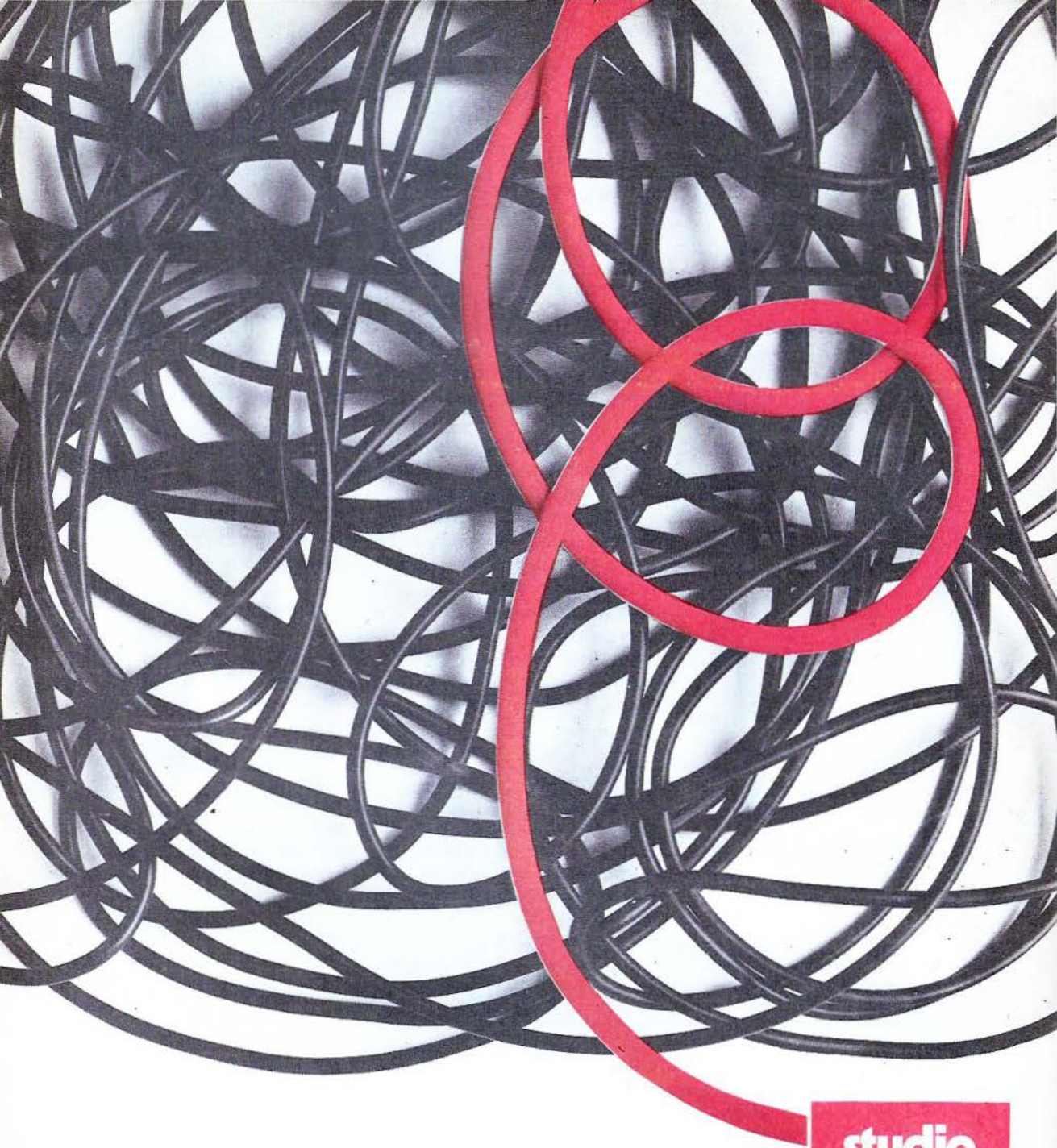
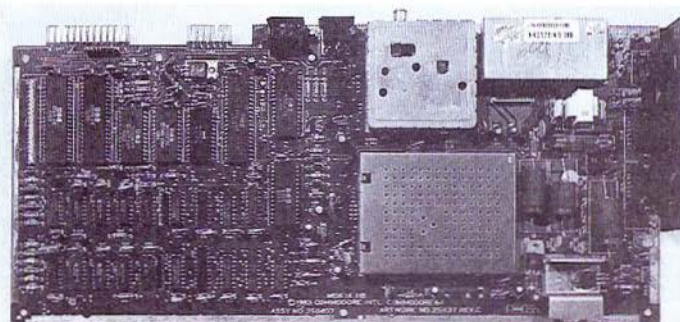
Queste locazioni contengono tutti i parametri relativi all'utilizzo dei canali aperti dalle routines KERNAL. Sono simili alle locazioni dalla 184 alla 186, soltanto che in questo caso vengono conservate le informazioni di tutti i canali aperti e non solo dell'ultimo. Lo spazio necessario è quindi di 30 bytes, considerando che il massimo numero di canali apribili contemporaneamente è 10.

(da 631 a 640) BUFFER DI TASTIERA

E' di tipo FIFO (first in, first out). Il primo carattere ad entrare è anche il primo ad uscire. La sua lunghezza è di dieci caratteri. Qui vengono tenuti fino all'utilizzo i codici ASCII dei tasti premuti. E' possibile usare questo buffer riempendolo con delle istruzioni di POKE, e aggiornando di conseguenza la locazione 198.

(da 641 a 644) LIMITI FISICI PER LA SCRITTURA DEI PROGRAMMI

Queste locazioni puntano rispettivamente al byte più basso e più alto della zona in cui è possibile scrivere programmi.

Questi valori sono automaticamente settati dal registro di controllo di I/O (locazione 1) durante l'accensione della macchina, e possono essere diversi se ad esempio avete una cartuccia inserita nella porta d'espansione. I valori di default sono 2048 e 40960. Non confondete quindi questi puntatori con quelli contenuti nelle locazioni dalla 43 alla 56.

HEX	DEC	DESCRIZIONE
00C5	197	**Tasto premuto attualmente.**
00C6	198	**Numero di caratteri presenti nel buffer di tastiera.**
00C7	199	**Interruttore per carattere in Reverse.**
00C8	200	**Pointer: fine del file logico per INPUT.**
00C9-00CA	201-202	**Posizione del cursore per l'INPUT.**
00CB	203	Flag: Caratteri Shiftati. Il valore di default é 64. Questa locazione mostra il valore del carattere nella 197 quando questo viene stampato sullo schermo
00CC	**204**	**Flag: Lampeggio cursore. 0=acceso, 1=spento.**
00CD	**205**	**Timer: conto alla rovescia per il lampeggio del cursore. Valore di default 2.**
00CE	**206**	**Codice ASCII del carattere nascosto dal cursore. Valore di default 32.**
00CF	**207**	**Flag: Ultimo lampeggio del cursore ON o OFF. Valore di default 0.**
00D0	208	Flag: segnalazione della presenza di INPUT o GET da tastiera. Valore di default 0.
00D1-00D2	209-210	Pointer: Indirizzo della linea corrente di schermo. E' difficile da usare.
00D3	211	Numero della colonna del cursore nella linea corrente.
00D4	212	Flag: 0= editor normale, qualsiasi altro valore=editor sotto virgolette.
00D5	**213**	**Lunghezza della linea fisica di schermo.**
00D6	214	Numero di linea fisica in cui sitrova il cursore. I valori normali variano da 0 a 24. E' possibile, ma difficile, controllare la posizione verticale del cursore POKE-ando valori da 0 a 255.
00D7	215	Area dati temporanei.
00D8	216	Flag: 0=editor normale, qualsiasi altro valore=modo di inserimento. Il valore esatto è il numero di inserimenti da sinistra.
00D9-00F2	**217-242**	**Link table per righe di schermo.**
00F3-00F4	243-244	Pointer: area colori schermo. Il valore di default è 55296.
00F5-00F6	245-246	Vector: Tavola di decodifica della tastiera.

HEX	DEC	DESCRIZIONE
00F7-00F8	247-248	Pointer: buffer di input per RS-232.
00F9-00FA	249-250	Pointer: buffer di uscita per RS-232.
00FB-00FE	251-254	Spazio libero. Questi bytes sono utilizzabili dal programmatore per indirizzamenti di tipo page-zero.
00FF	255	Area per dati temporanei del BASIC. Ultimo byte della pagina zero.
0100-010A	256-266	Area di lavoro per conversioni da virgola mobile a stringa.
0100-013E	256-318	Errore di input da nastro. E' parzialmente coperto dall'area precedente.
013F-01FF	319-511	Area usata dallo stack e da altre operazioni del sistema. Non usatela!!!!
0200-0258	512-600	Buffer per INPUT di sistema. Quando rispondete ad un INPUT mettete i dati in queste locazioni. Esse verranno vuotate appena possibile.
0259-0262	**601-610**	**Tavola KERNAL: numeri dei files logici attivi.**
0263-026C	**611-620**	**Tavola KERNAL: numero del device per ogni file.**
026D-0276	**621-630**	**Tavola KERNAL: indirizzo secondario per ogni file.**
0277-0280	**631-640**	**Buffer di tastiera.**
0281-0282	**641-642**	**Pointer: Indirizzo del punto più basso della memoria del sistema operativo. Valore di default 2048.**
0283-0284	**643-644**	**Pointer: Indirizzo del punto più alto della memoria del sistema operativo. Valore di default 40960.**

RESISTENTI, COMPATTE SILENZIOSE, EFFICIENTI, AFFIDABILI E COMPETITIVE.

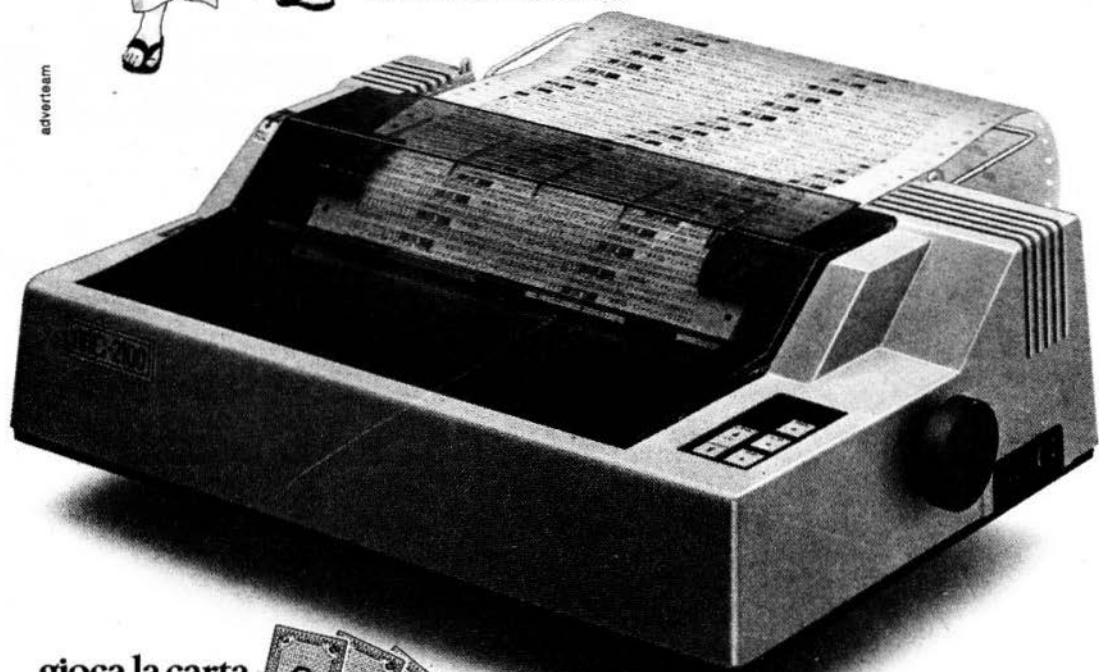
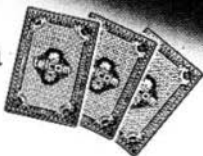
Stampanti MITSUI con le caratteristiche tipiche dei giapponesi.

La TELCOM propone una gamma di stampanti che si distingue per la varietà delle funzioni e per la grande affidabilità:

- 80 e 132 colonne
- 120 e 180 caratteri al secondo bidirezionale ottimizzata
- fogli singoli, moduli continui
- vari tipi di caratteri
- stampa espansa, compressa, NLQ, grafica
- interfaccia parallela, seriale, buffer fino a 128 KB, interfaccie speciali.

adverteam

gioca la carta telcom

Telcom s.r.l. 20148 Milano - Via M. Civitali, 75 - Tel. 4047648 (3 linee ric. aut.) - Telex 335654 TELCOM I

PROGRAMMAZIONE STRUTTURATA

di **Mariangela Guardione**

IV parte

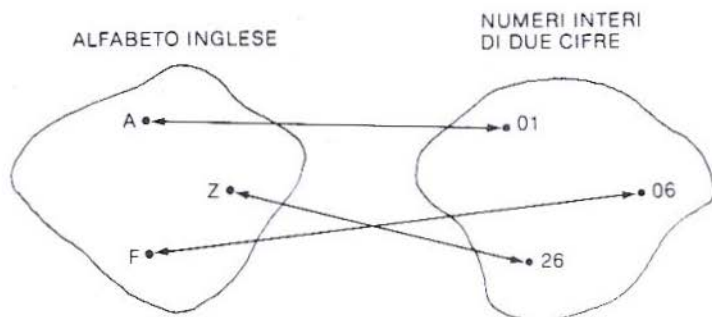



Fig. 1

Nella puntata precedente si è spiegato cosa sono i dati numerici, come sono rappresentati e quali sono le operazioni che si possono eseguire con loro.

Accanto a questi però, vi sono altri tipi di dati elementari: le Stringhe di Caratteri.

Questa struttura di dati è molto importante in quanto gran parte delle informazioni che sono trattate nei sistemi informatici non sono rappresentate dal tipo numerico. Per esempio i dati anagrafici delle persone, i codici fiscali o quelli bancari ed altre informazioni sono rappresentati da parole che possono essere lette, stampate, confrontate ed ordinate, ma non possono essere sommate o moltiplicate. Una "stringa di caratteri" è, quindi, una sequenza di simboli letterali che appartengono all'"alfabeto" riconosciuto dal computer. Ciascuno di questi simboli viene codificato (cioè rappresentato) nella memoria di un elaboratore tramite un numero o codice; uno dei codici più usati nell'ambito dei mini e personal computers per rappresentare questi caratteri è quello ASCII.

Ma che cos'è un codice? Esso è una corrispondenza fra due insiemi di simboli in modo tale che per definirlo è necessario che siano dati:

- i simboli che appartengono ai due insiemi
- le regole per far corrispondere biunivocamente a ciascun simbolo di un insieme, uno e uno solo dei simboli dell'altro.

Tutto questo è illustrato nella figura 1.

I testi composti di sole lettere si potrebbero rappresentare con testi numerici associando, quindi, alle lettere dell'alfabeto i numeri interi positivi di due cifre, a patto però di poter applicare a questo testo codificato l'asso-ciazione inversa allo scopo di poter riprodurre di nuovi il testo originale.

Come dicevamo precedentemente, il più diffuso di questi codici è quello ASCII che significa: American Standard Code For Information Interchange, ovvero "standard americano per codifica utile allo scambio delle informazioni".

Ciascuno dei suoi simboli è rappresentato con una sequenza di 7 bit.

Le due colonne di sinistra comprendono i caratteri di controllo utilizzati nelle applicazioni della trasmissione di dati. In realtà, nella memoria di un elaboratore per rappresentare un carattere vengono utilizzati quasi sempre 8 bit (1 byte); in ASCII le configurazioni in eccedenza sono qualche volta utilizzate per arricchire l'insieme dei segni grafici visualizzati sullo schermo come per esempio per la creazione di programmi per la gestione sia di "maschere" che di "testi".

Tutto questo viene illustrato nella figura 2.

Codice ASCII	Carattere	Codice ASCII	Carattere	Codice ASCII	Carattere
000	NUL	043	+	086	V
001	SOH	044	,	087	W
002	STX	045	—	088	X
003	ETX	046	.	089	Y
004	EOT	047	/	090	Z
005	ENQ	048	0	091	[
006	ACK	049	1	092	\
007	BEL	050	2	093	]
008	BS	051	3	094	Δ
009	HT	052	4	095	<
010	LF	053	5	096	'
011	VT	054	6	097	a
012	FF	055	7	098	b
013	CR	056	8	099	c
014	SO	057	9	100	d
015	SI	058	:	101	e
016	DLE	059	;	102	f
017	DC1	060	<	103	g
018	DC2	061	=	104	h
019	DC3	062	>	105	i
020	DC4	063	?	106	j
021	NAK	064	@	107	k
022	SYN	065	A	108	l
023	ETB	066	B	109	m
024	CAN	067	C	110	n
025	EM	068	D	111	o
026	SUB	069	E	112	p
027	ESCAPE	070	F	113	q
028	FS	071	G	114	r
029	GS	072	H	115	s
030	RS	073	I	116	t
031	US	074	J	117	u
032	SPACE	075	K	118	v
033	!	076	L	119	w
034	"	077	M	120	x
035	#	078	N	121	y
036	$	079	O	122	z
037	%	080	P	123	{
038	&	081	Q	124	|
039	'	082	R	125	}
040	(	083	S	126	~
041	)	084	T	127	DEL
042	*	085	U		

Posizione dei bit b4	b3	b2	b1	b7 b6 b5: 0 0 0	0 0 1	0 1 0	0 1 1	1 0 0	1 0 1	1 1 0	1 1 1
0	0	0	0	NUL	DLE	SP	0	@	P		p
0	0	0	1	SOH	DC1	!	1	A	Q	a	q
0	0	1	0	STX	DC2	"	2	B	R	b	r
0	0	1	1	ETX	DC3	#	3	C	S	c	s
0	1	0	0	EOT	DC4	$	4	D	T	d	t
0	1	0	1	ENQ	NAK	%	5	E	U	e	u
0	1	1	0	ACK	SYN	&	6	F	V	f	v
0	1	1	1	BEL	ETB	'	7	G	W	g	w
1	0	0	0	BS	CAN	(	8	H	X	h	x
1	0	0	1	HT	EM	)	9	I	Y	i	y
1	0	1	0	LF	SUB	*	:	J	Z	j	z
1	0	1	1	VT	ESC	+	;	K	[	k	}
1	1	0	0	FF	FS	,	<	L	\	l	|
1	1	0	1	CR	GS	-	=	M	]	m	{
1	1	1	0	SO	RS		>	N	^	n	~
1	1	1	1	SI	US	/	?	O	—	o	DEL

caratteri di controllo — caratteri grafici

gruppo ristretto dei caratteri ASCII

Significato dei simboli di controllo per trascrizione di dati

LAYOUT CHARACTERS
- bs backspace
- ht horizontal tabulation
- lf line feed
- vt vertical tabulation
- ff form feed
- cr carriage return

IGNORE CHARACTERS
- null null character
- can cancel
- sub substitute
- del delete

SEPARATOR CHARACTERS
- fs file separator
- gs group separator
- rs record separator
- us unit separator

ESCAPE CHARACTERS
- so shift-out
- si shift-in
- esc escape

MEDIUM CONTROL CHARACTERS
- bel ring bell
- dc1 -
- dc4 device control
- em end of medium

COMMUNICATION CONTROL CHARACTERS
- soh start of heading
- stx start of text
- etx end of text
- eot end of transmission
- enq enquiry
- ack acknowledgement
- nak negative acknowledgement
- dle data link escape
- syn synchronous idle
- etb end of transmission block

Fig. 2

Una sequenza di caratteri racchiusa tra doppi apici è una costante stringa, rappresentazione grafica è data dallo schema di figura n. 3.

Fig. 3

Una stringa che non contiene alcun carattere si chiama appunto "stringa vuota". Una variabile stringa può essere descritta, come già siamo abituati a vedere, con la seguente sintassi nelle figure n. 4.1 e 4.2.

Fig. 4.1

Fig. 4.2

Fig. 5

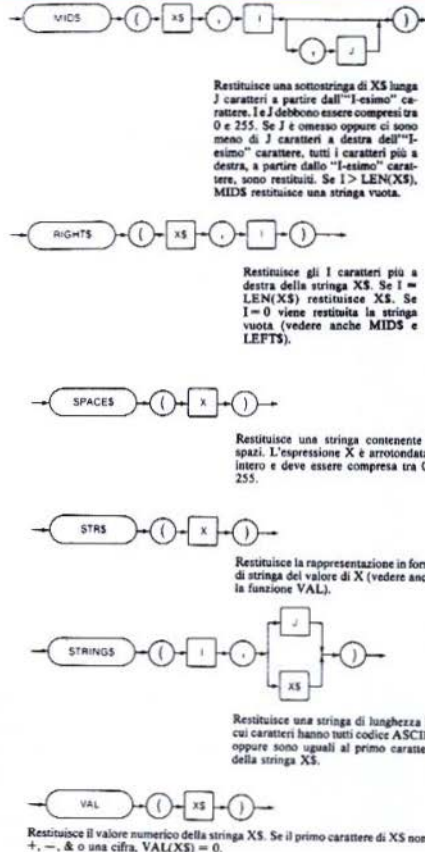
Fig. 5 (2ª parte)

Anche per le stringhe vi sono delle funzioni intrinseche del BASIC che operano su di esse. Qui di seguito le riportiamo, utilizzando per brevità di scrittura le seguenti convenzioni:

- X $, Y$, indicano le espressioni stringa
- I, J rappresentano le espressioni intere

Tutti i dati che abbiamo visto fino ad ora (numeri interi, reali e stringhe di caratteri) possono essere organizzati mediante "strutture di dati" che offrono notevoli possibilità da un punto di vista dell'applicazione.

Questa struttura viene definita tramite i suoi componenti, la sua organizzazione, le regole e gli operatori che servono per utilizzarla e gestirla.

Una delle più importanti fra queste strutture di dati, che viene utilizzata in quasi tutti i linguaggi di programmazione ad alto livello, è l'"array", detto anche "matrice" o "tabella".

Esso ha le seguenti proprietà:

- ogni componente dell'array ha un nome specifico ed è accessibile direttamente.
- Il numero degli elementi viene stabilito al momento dell'introduzione dell'array e non può essere più modificato.

Tutto questo rende necessario introdurre delle convenzioni che servono:

- per indicare le singole componenti di un array.
- Per specificare sia l'array che il numero delle sue componenti.

L'indicatore della componente di un array è rappresentato dal suo nome e dall'indice che individua in maniera univoca la componente.

Array a una dimensione

Un array a una dimensione è rappresentata da una corrispondenza tra gli insiemi dei numeri naturali ed

un insieme A di elementi tutti dello stesso tipo, cioè omogenei, nel senso che essi sono o tutti numeri interi oppure tutte stringhe. In una rappresentazione pratica si immagina un array come un insieme finito di "celle" contigue, che contengono ciascuna un dato di un certo tipo comune a tutte le celle. Tutto questo viene illustrato nella figura 6.

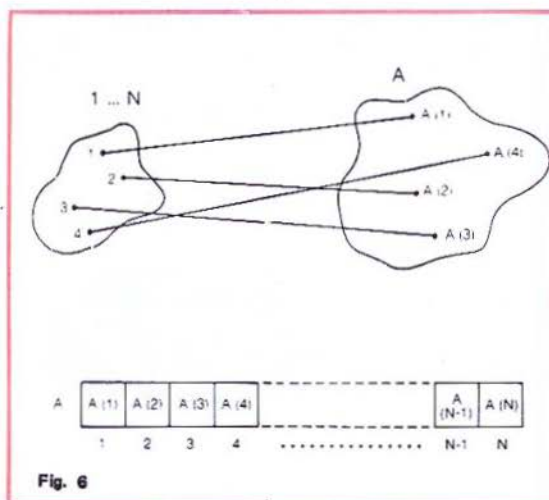



Fig. 6

Per poter accedere ad un suo elemento è necessario specificare il valore di un "indice". L'accesso in questo modo risulta essere diretto in quanto per individuare I-esimo elemento non è necessario considerare quelli con indice da 1 a I-1 oppure da I+1 ad N.

Una volta individuato l'elemento dell'array, questo può essere usato come una variabile, questo è il motivo per cui gli arrays molto spesso vengono chiamati anche "variabili con indice". Un metodo molto utilizzato nella pratica per la ricerca di una componente di un array ordinato è la procedura nota col nome di "ricerca binaria" che dà un esempio di accesso non diretto ai suoi elementi.

In questa procedura le componenti dell'array sono considerate come i nodi di un albero, in cui da ogni nodo si dipartono due rami (albero binario).

Il procedimento di calcolo di questo algoritmo è semplice. Si prende un intervallo di ricerca e si confronta la stringa contenuta a metà di detto intervallo con quella che si cerca. In questo modo si avrà uno dei seguenti tre risultati:

- la stringa di mezzo coincide con quella cercata e quindi l'esecuzione è terminata;
- la stringa cercata si trova nella prima metà dell'intervallo di ricerca; si eliminerà allora la seconda metà di detto intervallo, confrontando quindi la stringa desiderata con quella di mezzo al sottointervallo rimasto.
- La stringa cercata si trova nella seconda metà dell'intervallo di ricerca; si elimina la prima metà, e si confronta la stringa desiderata con quella di mezzo al sottointervallo rimasto.

La procedura continua finché non si è trovata la stringa desiderata. Questa metodologia di ricerca verrà utilizzata nel programma di questa puntata, in cui si eseguirà l'ordinamento e la ricerca di un array: verrà riportato anche, per meglio comprenderene il funzionamento, una rappresentazione grafica dell'algoritmo.

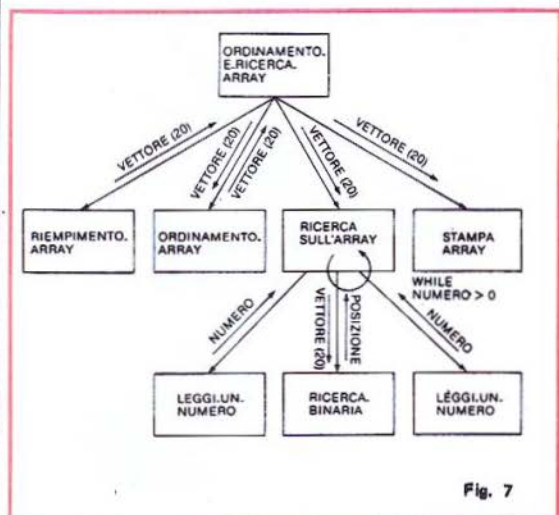



Fig. 7

Si passa ora a rappresentare la sintassi relativa agli arrays monodimensionali che è data nella figura n. 8.

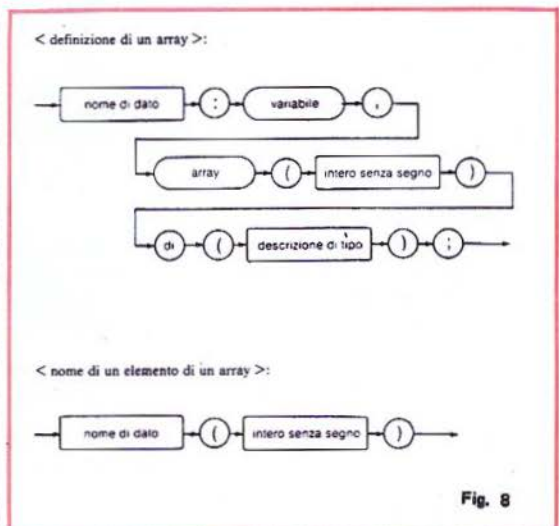



Fig. 8

Arrays a più dimensioni

In una struttura di questo tipo: <nome di dato>: variabile, array (<intero senza segno>).......... variabile di array (<descrizione di tipo>) in cui ricorre più di una volta, esempio N volte, la specifica <array...> si parla di array a una, due, ..., N dimensioni. La sintassi è rappresentata nella figura 9.

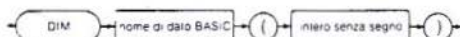



Fig. 9

Bisogna specificare che nella pratica gli array a una dimensione vengono chiamati "vettori" (variabili con indice) a quelli a due dimensioni "matrici" (variabili con due dimesioni che possono essere visti come un insieme di vettori).

A questi arrays a una o multi-dimensione si può associare, per meglio visualizzarli, delle immagini che possono essere un rettangolo e un parallelepipedo, come rappresentato in figura 10.

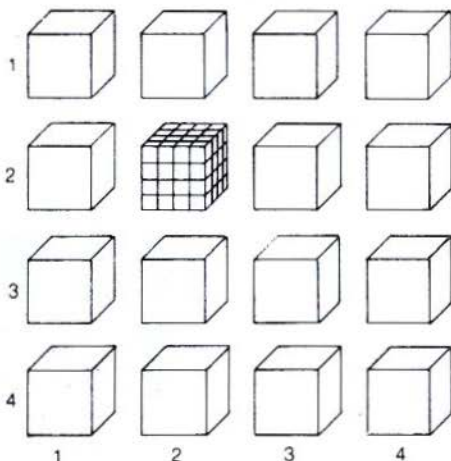



Fig. 10 (2ª parte)

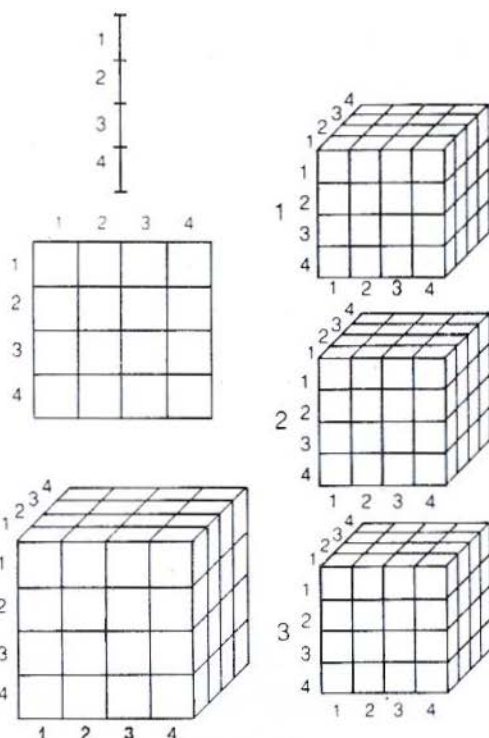



Fig. 10

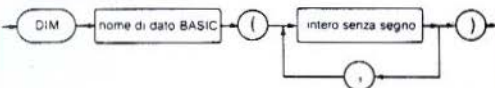
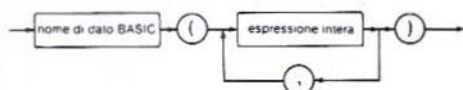



Fig. 10 (3ª parte)

Ogni insieme di indici individua un segmento, un quadratino o un cubetto che contiene un valore dell'array. Per meglio spiegare queste immagini illustriamo qui di seguito un esempio. Se consideriamo un ufficio in cui vi siano N*M raccoglitori che sono disposti per N righe e M colonne; e ciascuno di essi è costituito da J*K cassetti che sono disposti su J righe e K colonne; ciascun cassetto è suddiviso in scomparti che sono numerati da 1 a I a partire dalla maniglia. Se si vuole cercare, in quell'ufficio, una scheda bisogna specificare un valore per ciascuno degli indici N, M, J, K, I per poter accedere alla posizione della scheda cercata.

Trattamento degli arrays a più dimensioni

Il valore di ciascuna espressione è minore o uguale del corrispondente (cioè che si trova nella stessa posizione) intero senza segno nella lista degli indici. Se

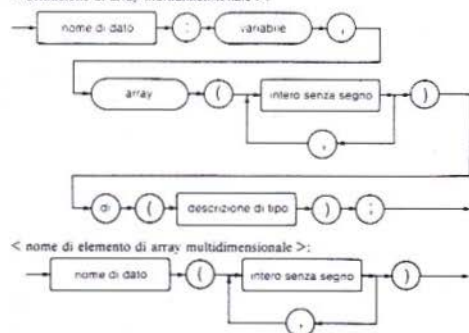



Fig. 11

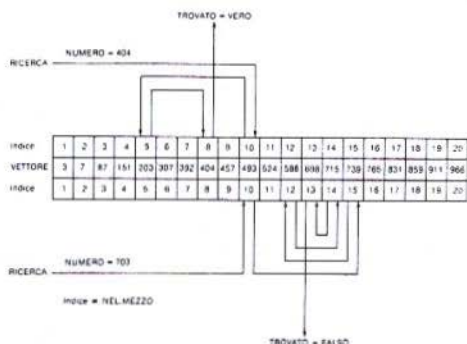



Fig. 12

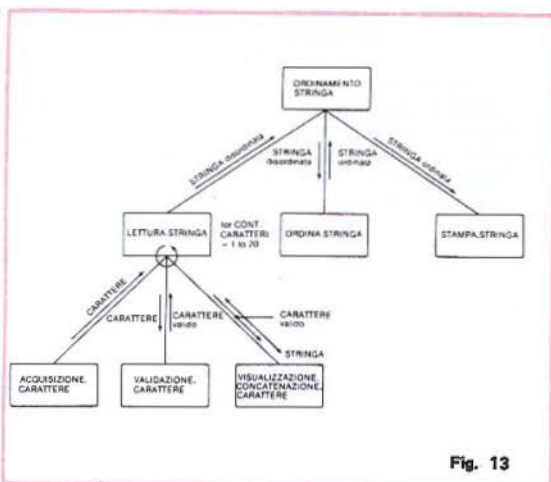



Fig. 13

viene usato un valore reale come indice, esso viene arrotondato ad un intero. Per concludere si riporta a seguito il listato del programma "ordinamento e ricerca in un array" e il relativo diagramma di struttura in cui si mostra come può essere definito un array e come possono essere utilizzati i suoi elementi. Infatti questo programma riempie un vettore con 20 numeri generati casualmente e che sono compresi fra 1 e 1000 e quindi li ordina in modo ascendente. Propone quindi all'utilizzatore di fornire un numero fra 1 e 1000, lo acquisisce, cerca nel vettore se è presente e quindi segnala l'esito della richiesta. Inoltre, se richiesto, visualizza il contenuto dell'intero vettore.

Questa ricerca viene eseguita utilizzando il metodo di cui abbiamo precedentemente parlato: la ricerca binaria o dicotomica.

```
C R O S S   R E F E R E N C E

PROGRAMMA : BIN

------*---------------- - - -
VAR.  ! LINEA DEL PROGRAMMA
------*---------------- - - -
      !
A     !  265
FA    !  195
IN    !  215   220   225   245   250   255
      !  265   275   385   390   395
NM    !  505   510   520   525   530
NU    !  330   335   340   440   445   450
      !  460   510   525
PD    !  250   255   265   270
PO    !  335   490   520
PR    !  490   495   505   530
S$    !  345   375   380   400
TR    !  490   520
UL    !  490   495   505   525
VE    !  195   520
VT(   !  195   220   255   265   390   510
      !  525
```

```
100 REM ************************
105 REM *ORDINAMENTO E RICERCA *
110 REM *      IN UN ARRAY     *
115 REM *          DI          *
120 REM * MARIANGELA GUARDIONE *
125 REM ************************
130 REM *DESCRIZIONE DEI DATI  *
135 REM *DATI IN INGRESSO:     *
140 REM *NU=NUMERO CASUALE     *
145 REM *DATI IN USCITA        *
150 REM *VT=VETTORE DEI NUMERI *
155 REM *   CASUALI            *
160 REM *PO=INDICE DEL VETTORE *
165 REM *DATI AUSILIARI        *
170 REM *IN=INDICE             *
175 REM *PD=INDICE             *
180 REM ************************
185 REM *  ELABORAZIONI DATI   *
190 REM ************************
195 DIM VT(20):FA=0:VE=-1
200 REM ************************
205 REM *   RIEMPIMENTO ARRAY  *
210 REM ************************
215 FOR IN=1 TO 20
220 VT(IN)=INT(1000*RND(1000)+.5)
225 NEXTIN
230 REM ************************
235 REM *   ORDINAMENTO ARRAY  *
240 REM ************************
245 FOR IN=1 TO 19
250 FOR PD=IN+1 TO 20
255 IF VT(PD)<VT(IN) THEN 265
260 GOTO 270
265 A=VT(IN):VT(IN)=VT(PD):VT(PD)=
    A
270 NEXTPD
275 NEXTIN
280 REM ************************
285 REM *  RICERCA SUGLI ARRAY *
290 REM ************************
295 PRINT"[CLEAR][3 DOWN]"
300 PRINT"ORDINAMENTO E RICERCA AR
    RAY"
305 GOSUB  425
310 REM ************************
315 REM *   INIZIO PROCEDURA   *
320 REM *   RICERCA BINARIA    *
325 REM ************************
330 IF NU=0 THEN 360
335 GOSUB 490:IF PO>0 THEN PRINT"N
    UMERO";NU;"TROVATO IN POSIZION
    E";PO:GOTO 350
340 PRINT"NUMERO";NU;"NON TROVATO"
345 GET S$:IF S$="" THEN 345
350 GOSUB 425
355 GOTO 330
360 REM ************************
365 REM *    STAMPA ARRAY      *
370 REM ************************
375 INPUT "DESIDERI LA STAMPA DELL
    'ARRAY(S/N)";S$
380 IF S$="N" THEN 450
385 FOR IN=1 TO 20
390 PRINTVT(IN);"   ";
395 NEXTIN:PRINT
400 GET S$:IF S$="" THEN 400
405 GOTO 545
410 REM ************************
415 REM *  LETTURA DEL NUMERO  *
420 REM ************************
425 PRINT"INDOVINARE UNO DEI 20 NU
    MERI"
430 PRINT"COMPRESI FRA 1 E 1000"
435 PRINT"(0 PER TERMINARE)"
440 INPUT NU
445 IF NU=0 THEN 465
450 IF NU>=1 AND NU<=1000 THEN 470
455 PRINT"[DOWN]RIPETERE IL NUMERO
    "
460 INPUT "COMPRESO FRA 1 E 1000";
    NU
465 RETURN
470 REM ************************
475 REM *   PROCEDURA RICERCA  *
480 REM *       BINARIA        *
485 REM ************************
490 TR=0:PR=1:UL=20:PO=0
495 IF PR<=UL THEN 505
500 GOTO 540
505 NM=INT((PR+UL)/2+.5)
510 IF NU=VT(NM) THEN 520
515 GOTO 525
520 TR=VE:PO=NM:GOTO 540
525 IF NU<VT(NM) THEN UL=NM-1:GOTO
     535
530 PR=NM+1
535 GOTO 495
540 RETURN
545 END
```

STATISTICA

di **Mariangela Guardione**

seconda puntata

In questa seconda puntata continuiamo ad esaminare le variabili più importanti della Statistica.
Come si è già visto nel numero precdente, le medie servono ad individuare, un valore "centrale" di un insieme di dati. Esse, peraltro, non forniscono ulteriori informazioni su altri aspetti dell'insieme considerato.

Per far questo è necessario introdurre un concetto: "la variabilità o dispersione" dei dati, la cui importanza non è certamente inferiore a quella del "valore medio". Quando si parla di "dispersione" s'intende l'attitudine che certi dati numerici hanno a disporsi intorno ad un valore medio.
Una distribuzione di dati è caratterizzata da altre due grandezze che sono: la Mediana e la Moda.

Mediana

La Mediana di un insieme di numeri ordinati in grandezza è il valore centrale oppure la media aritmetica di due valori centrali. Per esempio: l'insieme dei numeri 3, 4, 5, 6, 7, 7, 7, 9 ha mediana 6 poiché questo numero rappresenta il valore centrale della nostra distribuzione. Ed ancora: l'insieme dei numeri 5, 5, 7, 9, 11, 12, 15, 18 ha mediana (9+11) /2=10 poiché 9 e 11 sono i valori centrali della distribuzione.

Moda

La Moda di un insieme di numeri è il valore che si presenta con la più alta frequenza, cioè il valore più comune. La moda può non esistere, e in caso contrario può non essere unica. Ad esempio: l'insieme dei numeri 2, 2, 5, 7, 9, 9, 9, 10, 10, 11, 12, 18 ha moda 9. Ed ancora: l'insieme dei numeri 3, 5, 8, 10, 12, 15, 16 non ha moda.

Due distribuzioni, quindi, possono avere uguali la media aritmetica, la moda, la mediana e tuttavia essere molto diverse. Si osservi, a questo proposito i due diagrammi riportati nella figura 1.

In aritmetica i casi, come si può osservare, la media aritmetica, la moda e la mediana sono uguali a 10. Se si esaminano però attentamente i due grafici, si possono osservare delle profonde differenze tra le due distribuzioni. Infatti i dati sono diversamente dispersi, in quanto i primi sono compresi tra 6 e 14, mentre i secondi tra 2 e 18. A questo punto, bisogna puntualizzare alcuni concetti: i fenomeni collettivi sono costituiti da un insieme (o meglio da una pluralità) di casi singoli, che sono chiaramente differenti fra di loro; è per questo motivo che si considerano i fenomeni collettivi come variabili; la loro attitudine ad assumere diverse modalità qualitative o quantitative si chiama "variabilità". Quindi accanto ai valori medi in Statistica si considerano altre grandezze il cui utilizzo serve ad evidenziare il maggiore o minore grado di variabilità del fenomeno.

Queste grandezze sono note come "indici di variabilità". Si dividono in: indici assoluti, indici relativi.

Gli indici assoluti sono espressi utilizzando la stessa unità di misura del fenomeno a cui si riferiscono. Gli indici relativi, invece, rappresentano i rapporti tra grandezze omogenee di uguale unità di misura e in quanto tali, essi sono adimensionali, cioè sono dei "numeri puri".

Gli indici relativi sono utilizzati quando si deve effettuare un confronto fra le variabilità di fenomeni che sono espressi in unità di misure diverse, oppure tra più fenomeni di diverso ordine di grandezza, espressi nella stessa unità di misura.

Quando si esamina la variabilità di un fenomeno bisogna prima di tutto precisare quale aspetto di essa si vuole considerare, poiché la "variabilità" è un concetto (e quindi non è misurabile), mentre si può valutare qualche suo particolare modo di manifestarsi. Ciascun indice di variabilità, però, misura solo un aspetto di essa, e ha quindi una validità limitata. Si può quindi verificare che la distribuzione indicata con A risulti più variabile della distribuzione B in base a un indice, e invece meno variabile rispetto ad un altro; oppure ancora che, secondo un certo indice, il fenomeno A presenti un aumento della variabilià dal tempo 1 al tempo 2, e abbia invece una diminuzione rispetto ad un altro.

Queste discordanze non devono stupire: ciascun indice ha tutte le proprietà strutturali che servono per caratterizzare particolari aspetti di un fenomeno, che in quanto tali possono diffe-

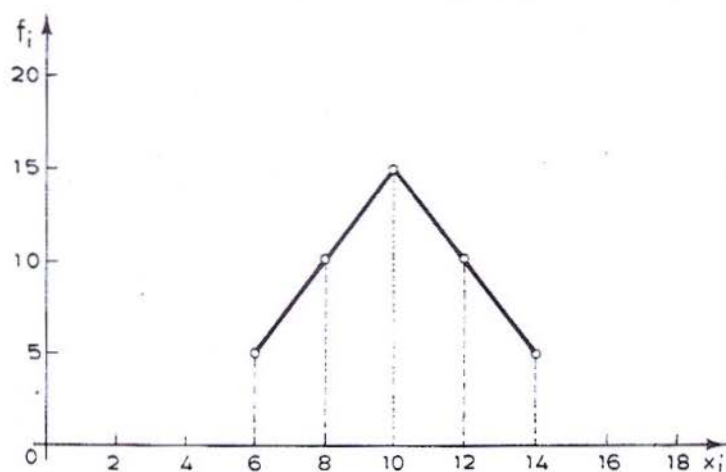
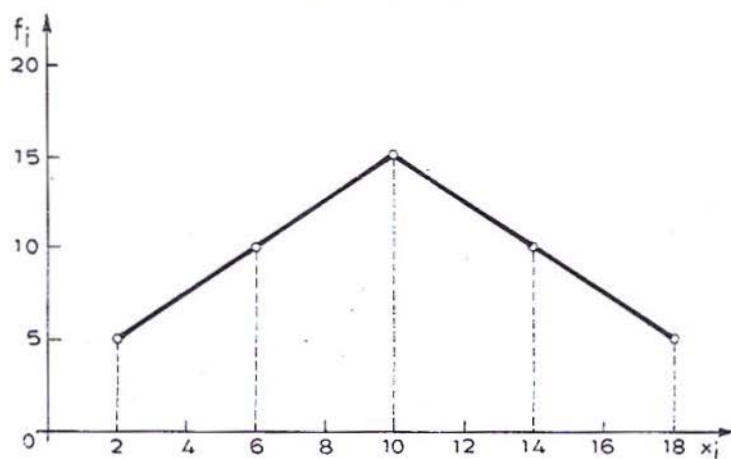
x_i	f_i
6	5
8	10
10	15
12	10
14	5

x_i	f_i
2	5
6	10
10	15
14	10
18	5

Fig. 1

rire tra loro.

In questa puntata, quindi, esamineremo alcuni indici di variabilità.

Campo di variazione

Questo è un indice assoluto, e rappresenta la differenza tra il valore massimo e quello minimo in una distribuzione di dati. Il campo di variazione, però, dà una misura grossolana della distribuzione dei dati, poiché non considera minimamente i dati che sono compresi tra il valore minimo e massimo. Un esempio: Il campo di variazione dei numeri 2, 3, 3, 7, 8, 15, 13 è 15—2=13.
Spesso il campo di variazione è dato semplicemente indicando il più piccolo e il più grande della serie dei numeri dati.
Nel caso precedente il campo di variazione è 2—12.

Deviazione standard

Scostamento Medio Assoluto dalla Media Aritmetica. Questo indice assoluto rappresenta, come d'altra parte espresso dal nome, la media aritmetica degli scostamenti presi in valore assoluto dalla media aritmetica.

La sua espressione matematica è data dalla sommatoria del valore assoluto delle differenze tra ciascun dato e la sua media, il tutto diviso per il numero dei dati.

Ad esempio: dobbiamo trovare lo scostamento medio assoluto dei numeri 2, 4, 5, 6, 8.

Media aritmetica=X=(2+4+5+6 +8) /5=5

Scostamento medio assoluto
|2—5|+|4—5|+|5—5|+|6—5|+| 8—5| /5=
(|—3|+|—1|+|0|+|1|+|+3| /5=
3+1+0+1+3)/5 = 8/5 = 1.6

Se si rappresentano questi risultati in un sistema di assi cartesiani ortogonali, si pone sull'asse delle ordinate le modalità x, mentre sull'ascissa i dati, come nella figura 2.
Se si studia questo diagramma, si osserva che lo scostamento medio assoluto della media X non è che la media aritmetica delle distanze espresse dai segmenti tratteggiati, fra ciascun dato e la loro media aritmetica.

Scarto quadratico medio e varianza

Lo scarto quadratico medio di un insieme di N numeri viene indicato con s., che rappresenta la radice quadrata della media dei quadrati degli scarti dalla media aritmetica.

A volte quando si calcola lo scarto quadratico medio dei dati di un campione nel denominatore al posto di N viene sostituito (N—1), poiché il valore risultante rappresenta una migliore stima dello scarto quadratico medio della popolazione. Per grandi valori (almeno N>30) non sussiste alcuna differenza tra le due definizioni.

La varianza, invece, di un insieme di dati è definita come il quadrato dello scarto quadratico medio ed è indicata con s 2. Per concludere si riporta il listato del programma con cui si può calcolare lo scarto quadratico medio e la varianza per una qualsivoglia distribuzione di numeri.

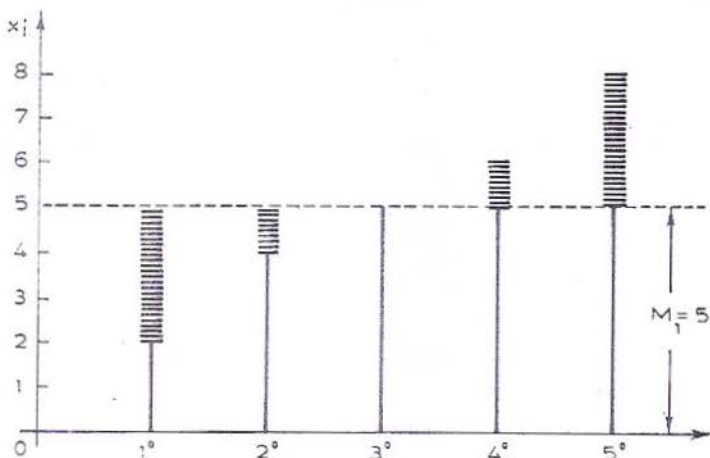



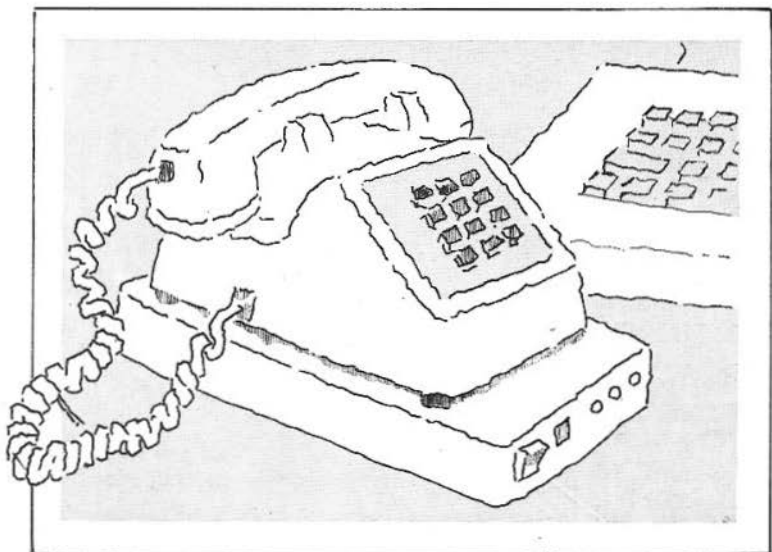
Fig. 2

```
C R O S S   R E F E R E N C E

PROGRAMMA : DSTA

--------*------------------- - - -
VAR.    ! LINEA DEL PROGRAMMA
--------*------------------- - - -
        !
A       !   600   610   615
A2      !   590   615   630
B$      !   340   345   350
C$      !   420   425   430   475
C(      !   160   165   275   320   450
F(      !   160   165   270   315   435   445
FF      !   270   280   285   290
I       !   165   270   275   300   315   320
        !   595   600   620
J       !   315   320   405   435   445   450
J$      !   400   405
M       !   410   435   445   495   590   610
        !   625   630   635   640
N       !   190   240   255   315   595   630
N1      !   590   605   625   630
N3      !   435   445   460
S       !   255   315   320   435   445
S$      !   295   655
S1      !   240   250   255
S2      !   240   635
S3      !   255   270
V       !   630   635   645   650
X       !   285   290   635
Y       !   515   520
```

```
100 REM *************************
105 REM *    STIMA STATISTICA   *
110 REM *          DI           *
115 REM *  MARIANGELA GUARDIONE *
120 REM *************************
125 PRINT"[CLEAR][2 DOWN][5 RIGHT]
    TEORIA DI STIMA STATISTICA"
130 REM *************************
135 REM * DIMENSIONAMENTO ARRAYS *
140 REM * E CICLO LETTURA VALORI *
145 REM *  DI AFFIDABILITA' E   *
150 REM *    SCARTO RELATIVO    *
155 REM *************************
160 DIM C(7),F(7)
165 FOR I=1 TO 7:READ C(I),F(I):NE
    XTI
170 REM *************************
175 REM * ROUTINE INPUT DEI DATI *
180 REM *************************
185 PRINT"[2 DOWN][2 RIGHT]INTRODU
    CI IL NUMERO DI CAMPIONI"
190 INPUT "[2 RIGHT]CONSIDERATI";N
195 REM *************************
200 REM *  RICHIAMO SUBROUTINE  *
205 REM *    IMMISSIONE DATI    *
210 REM *************************
215 GOSUB 565
220 REM *************************
225 REM *  BLOCCO DI CALCOLO E  *
230 REM * PRESENTAZIONE RISULTATI*
235 REM *************************
240 S1=S2*N/(N-1)
245 PRINT"[DOWN]STIMA NON DEVIATA
    DEL SIGMA QUADRO"
250 PRINT"VARIANZA DELLA POPOLAZIO
    NE =";S1
255 S=SQR(S1):S3=S/SQR(N)
260 PRINT"[DOWN]INTERVALLI DI AFFI
    DABILITA'"
265 PRINT"PER LA MEDIA[DOWN]"
```

```
270 FOR I=1 TO 7:FF=F(I)*S3
275 PRINT"LIVELLO DI AFFIDABILITA'
    = ";C(I)
280 PRINT"PIU' O MENO: ";FF
285 PRINT"MASSIMO= ";X+FF
290 PRINT"MINIMO= ";X-FF
295 GET S$:IF S$="" THEN 295
300 NEXTI
305 PRINT"[DOWN]INTERVALLI DI AFFI
    DABILITA' STIMATI"
310 PRINT"PER LA DEVIAZIONE STANDA
    RD:[DOWN]"
315 FOR I=1 TO 7:J=F(I)*S/SQR(2*N)
320 PRINT"  ";C(I),J,S+J,S-J:NEXTI
325 PRINT"[DOWN]SI VUOLE UN CALCOL
    O DELLA LARGEZZA"
330 PRINT"DEL CAMPIONE CHE SI DEVE
     PRENDERE"
335 PRINT"PER RIDURRE L'ERRORE DI
    STIMA"
340 INPUT "AD UNA QUANTITA' MASSIM
    A(S/N)";B$
345 IF B$="N" THEN 500
350 IF B$<>"S" THEN PRINT"[4 UP]":
    GOTO 325
355 PRINT"[DOWN]INTRODURRE IL LIVE
    LLO DI AFFIDABILITA' SCELTO";
360 PRINT"(FRA LE SCELTE PRECEDENT
    I)"
365 PRINT"[DOWN]   1 PER 50"
370 PRINT"   2 PER 60"
375 PRINT"   3 PER 70"
380 PRINT"   4 PER 80"
385 PRINT"   5 PER 90"
390 PRINT"   6 PER 95"
395 PRINT"   7 PER 99"
400 GET J$:IF J$="" THEN 400
405 J=VAL(J$)
410 INPUT "INTRODURRE IL MASSIMO E
    RRORE DI STIMA  DESIDERATO";M
415 PRINT"E' IN ESAME LA MEDIA(M)
    O LA "
420 INPUT "DEVIAZIONE STANDARD(S)"
    ;C$
425 IF C$="S" THEN 445
430 IF C$<>"M" THEN PRINT"[2 UP]":
    GOTO 415
435 N3=INT((S*F(J)/M)↑2)+1
440 GOTO 450
445 N3=INT(((F(J)*S/M)↑2)/2)+1
450 PRINT"[DOWN]AL LIVELLO DI AFFI
    DABILITA' PERCENTUALE DI";C(J)
455 PRINT"SAREBBE OPPORTUNO ASSUME
    RE UN CAMPIONE"
460 PRINT"DI";N3
465 PRINT"PER ESSERE SICURI CHE LA
     STIMA SULLO"
470 PRINT"ERRORE RELATIVO ALLA ";
475 IF C$="S" THEN 490
480 PRINT"MEDIA"
485 GOTO 495
490 PRINT"DEVIAZIONE STANDARD"
495 PRINT"NON SUPERI";M
500 PRINT"VUOI RIPETERE CON:"
505 PRINT"                     ALTRI
    CALCOLI     (1)"
510 PRINT"                    LO STE
    SSO CMPIONE(2)"
515 INPUT "                    O TER
    MINARE      (3)";Y
520 ON YGOTO 185,325,560
525 DATA  50,0.6744902454373
530 DATA  60,0.9416214285714
535 DATA  70,1.0364335334476
540 DATA  80,1.2815515569516
545 DATA  90,1.6448536821705
550 DATA  95,1.9599641025641
555 DATA  99,2.575827586207
560 END
565 REM ***********************
570 REM * SUBROUTINE IMMISIONE *
575 REM *     DATI E CALCOLO    *
580 REM *  MEDIA E VARIANZA     *
585 REM ***********************
590 M=0:N1=0:A2=0
595 PRINT"[DOWN]":FOR I=1 TO N
600 PRINT"[2 RIGHT]INSERIRE IL DAT
    O";I;:INPUT ":    ";A
605 N1=N1+1
610 M=M+A
615 A2=A2+A*A
620 NEXTI
625 M=M/N1
630 V=(A2-N1*M*M)/(N-1)
635 X=M:S2=V
640 PRINT"[DOWN][2 RIGHT]MEDIA=
         ";M
645 PRINT"[2 RIGHT]VARIANZA=
         ";V
650 PRINT"[2 RIGHT]DEVIAZIONE STAN
    DARD= ";SQR(V)
655 GET S$:IF S$="" THEN 655
660 PRINT"[CLEAR]":RETURN
```

L'AGENDA TELEFONICA AUTOMATICA

di **Ernesto Sidoti**

Sul numero 3 di questa rivista ho presentato una scheda elettronica capace di pilotare 8 relé. Questo mese vi propongo un'altra interessante applicazione della porta utente: un combinatore telefonico.
Tengo a precisare che questo tipo di realizzazione deve essere intesa solamente dal punto di vista sperimentale in quanto, se si legge il regolamento SIP, risulta vietato qualsivoglia manipolazione della rete telefonica.
La parte elettronica da applicare esternamente non è altro che 1/8 della scheda presentata tempo fa. Questa volta l'uscita utilizzata è una sola: la PB7; l'alimentazione — dato l'esiguo assorbimento del circuito — è presa dallo stesso computer. Il software accoppiato è una versatile agenda telefonica capace di archiviare, ricercare e modificare non solo nome, cognome e numero di telefonico, ma anche il completo indirizzo e una riga di commento. Con questa agenda si potranno memorizzare i 198 numeri più usati della agenda. Quando si vorrà telefonare basterà ricercare il nominativo, alzare il microtelefono e aspettare che il computer componga il numero telefonico. Se la linea risultasse occupata, niente paura! Basta premere un qualsiasi tasto del C64 per ripetere il numero.
Per meglio capire come può funzionare un combinatore telefonico computerizzato occorre far riferimento al suo diretto progenitore: il combinatore meccanico. Per intenderci, il disco trasparente con in numeri. Quando si compone una cifra, sul telefono si interrompe (si apre cioè) il circuito, per un numero di volte pari alla cifra composta, eccetto per lo O che genera 10 interruzioni. Queste interruzioni fanno commutare la linea su quella dell'utente desiderato. Le interruzioni devono avere un tempo ben preciso, così come le pause tra un interruzione e l'altra e le pause tra un numero ed un altro.
I tempi da rispettare sono: 40 mS per l'interruzione, 60 mS per la pausa tra un'interruzione e l'altra, 500mS invece la pausa tra un numero e l'altro. Quest'ultima è chiamata pausa interdigitale.
Faccio un esempio (osservate la figura n. 1). Se la prima cifra da comporre è un 2, il combinatore telefonico interromperà la linea per 40 mS, poi la ripristinarà per altri 60 e nuovamente la interromperà per 40. Infine la ripristinerà per 500 mS prima

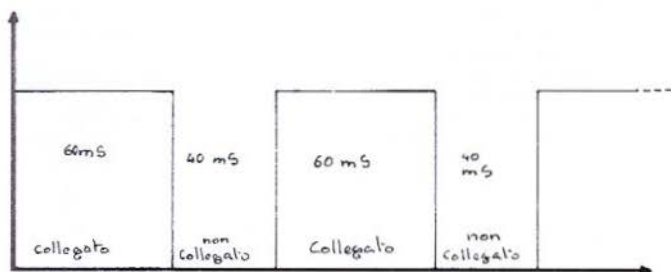
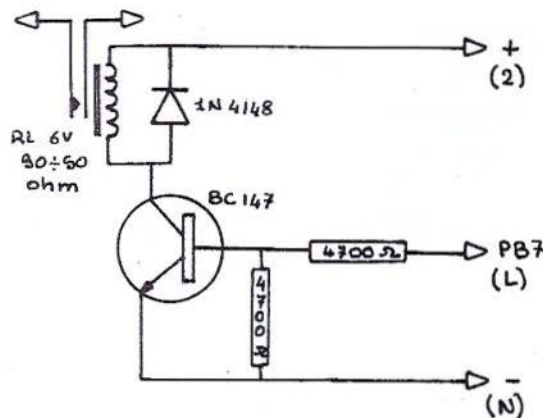
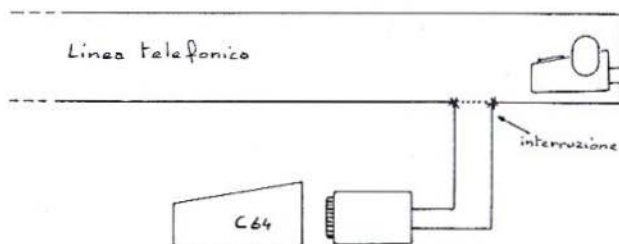
di comporre la cifra successiva.
Un combinatore computerizzato dovrà rispettare tali modalità. Il modo di realizzarlo è molto semplice. È sufficiente un relé, che non è altro che un interuttore comandato elettricamente, che collegato opportunamente alla linea telefonica interrompe la continuità della linea stessa rispettando i tempi sopra citati.
Tengo nuovamente a ricordare che questo circuito è presentato solo a scopo didattico in quanto qualsiasi manomissione della linea telefonica è vietata. (uomo avvisato mezzo salvato).

Installazione e Hardware

La scheda è formata da un semplice transistor che fa da servo/interruttore elettronico per il relé. La base di questo transistor è collegata tramite una resistenza alla pista della porta utente contraddistinta con la lettera L. Ogni qualvolta su questa pista si presenterà una tensione alta si ecciterà il relé collegato alla linea telefonica generando una interruzioni su di essa.
Per installare la scheda occorre interrompere uno dei due fili che vanno alla presa telefonica e quindi è necessario collegarlo a un terminale normalmente chiuso del relé e un secondo filo tra l'altro terminale del relé e il punto da dove hai scollegato la linea. (per ogni difficoltà di apprendimento osservate la figura 2).
Il costo di tutta la scheda difficilmente supererà le 12000 lire e l'allacciamento non più di 500 lire.

Il software

Il programma è estremamente semplice da usare grazie alla presenza di numerose indicazioni per i comandi disponibili. Il completo controllo degli input limita le possibilità di errore da parte dell'utente. È importante ricordare che prima di spegnere il computer occorre, se ci sono state modifiche, aggiornare il file. Altra raccomandazione è di scrivere sempre il recapito telefonico preceduto dal prefisso e da una barra trasversale (esempio 02/3245689).
Se non si conosce il prefisso può essere utile inserire degli zeri in attesa di sostituirli con il prefisso esatto. (esempio 00/3245698). Il programma funziona sia con il registratore che con l'unità disco, proprio per favorire una più larga fascia di lettori.
Linea **110**. Con POKE 650, 128 abilito il repeat su tutti i tasti, con POKE 56579, 128 setto in output la porta PB7 e porto a off la porta con POKE 56577,0.
Linea **1120**. Con POKE 54296, 15 definisco il volume della nota generata se si verifica una condizione di errore.
Linea **4800**. Eseguo la routine di inizializzazione generando un riquadro con la routine alla riga **1870**.
Linea **4810**. Eseguo una routine di posizionamento del cursore alla linea 7.
Linea **4820-4860**. Scelta disco o cassetta.
Linea **4870**. Se i tasti premuti sono diversi da N o S, va in esecuzione la routine sound e che genera la nota di errore,
Linea **4880-4990**. Scelgo se creare un nuovo archivio o leggere da periferica i vecchi dati. Se si sceglie di leggere il vecchio archivio digitando N si manda in esecuzione una delle due routine di lettura, una riguarda il registratore l'altra il disco. Terminata la routine di inizializzazione si visualizza i menù.
Linea **1200-1500**. Stampa menù principale e attende il comando da eseguire.
Linea **2650**. Se nel menù si sceglie l'opzione "memorizzare" si avvia la routine che archivia un nuovo utente.
Linea **5570-5650**. Per prima cosa si dovrà trovare il posto libero sull'array. Se è la prima memorizzazione ovviamente la PO, che è la variabile dove viene memorizzato il posto libero, sarà 1. Se invece c'é già stata qualche cancellazione la PO risulterà eguale al posto che occupava prima l'utente cancellato. L'algoritmo riconosce che il posto è vuoto se trova un asterisco nel terzo campo del record. Se non vi sono asterischi vuol dire che il posto deve essere in fondo all'array e quindi PO sarà eguale al primo record trovato vuoto. Se durante questa ricerca si supera il numero massimo dei record vuol dire che non esistono più spazi vuoti in memoria, viene quindi stampata la scritta AGENDA PIENA e si torna al menù.
Linea **2660-2740**. Quando invece si trova un valore di PO si avvia il processo di input dei dati riguardante l'utente. Sarà visualizzata una maschera e avviate le routine di input per ciascun campo del record.
Linee **2750-2940**. Appena finiti gli input l'algoritmo domanda se i dati appena acquisiti sono corretti, rispondendo no, digitanto RETURN, verrà visualizzata sulla parte inferiore dello schermo un piccolo menù indicante i diversi campi da correggere, digitando una delle opzioni indicate si potrà correggere il campo errato. Ancora una volta l'algoritmo chiede se i campi sono esatti e se si digita la barra (SPACE) si avvia l'algoritmo che sistema in ordine alfabetico l'utente entrato.
Linee **3000-3130**. Per evitare che durante la fase di ordinamento si facessero scambi di interi record, con la conseguente notevole perdita di tempo, l'ordinamento è fatto su un vettore indice. Ogni elemento del vettore indice (X (i)) contiene la reale posizione di un elemento dell'array. Se ad esempio sono stati memorizzati dieci nominativi nell'array, questi si troveranno nell'ordine con il quale sono stati immessi, ma con questo non è detto che siano alfabeticamente a posto.
Per i nostri usi è importante disporre di una lista ordinata per facilitare la ricerca. L'ordine alfabetico quindi è memorizzato su di un vettore, che avrà come primo elemento il numero del record con minor peso, nel decimo elemento quello con maggior peso e infine nei restanti elementi 0. Quindi per vedere il nome con minor peso non si dovrà visualizzare il primo elemento dell'array, ma bensì l'elemento NO$ (X (1), 1).
Dopo questo piccolo esempio torniamo all'algoritmo di rinnovo puntatori. All'ingresso dell'algoritmo troviamo un IF che testa se la prima posizione del vettore indice è libera, se così è allora vuol dire che ci troviamo nel caso di array vuoto quindi il nominativo in ingresso dovrà per forza avere indice 1 e posto di array 1. Se la condizione appena citata è falsa allora si memorizza l'intero record nella posizione PO precedentemente determinata, in seguito si avvia un ciclo che va dal primo elemento dell'array fino all'ultimo utilizzato (CL). All'interno di questo ciclo si testa l'esistenza di un nominativo di peso maggiore dell'ultimo entrato. Se c'è, allora si avvia un secondo ciclo che porta tutti i puntatori una posizione più in basso e inserisce nella posizione dove si trovava il puntatore al nominativo maggiore dell'ultimo entrato, il puntatore dell'ultimo nominativo.
Se invece non esistesse alcun nominativo con peso maggiore dell'ultimo entrato abbiamo tre possibilità. La prima è soddisfatta se è libera nell'array la posizione 1, in questo

caso il puntatore assumerà valore 1. La seconda scelta è soddisfatta se l'ultimo nominativo entrato ha peso maggiore di tutti quelli presenti nell'array. In questo caso il puntatore avrà valore eguale a CL+1. La terza scelta è soddisfatta se l'ultimo posto libero è proprio l'ultimo possibile, ossia MA e quindi il puntatore avrà valore CL. Eseguita questa routine si torna al menù principale.
L'altra scelta possibile da menù riguarda la ricerca che si articola in tre fasi distinte. La prima cerca realmente il nominativo, la seconda visualizza l'intera sequenza dei campi riguardanti l'utente cercato e da la possibilità di modificarne i contenuti o cancellare l'utente completamente, la terza compone automaticamente il numero telefonico.
Linee **3810-3950**. Scegliendo da menù <F3> si entra nell'opzione ricerca. Come prima operazione l'algoritmo chiede con quale lettera comincia il cognome ricercato. Se il tasto digitato è una lettera dell'alfabeto si avvia la ricerca. La ricerca è suddivisa a gruppi di undici nominativi. Se la lettera cercata esiste nell'array si esce dal ciclo e si passa alla parte visualizzazione della ricerca. Se la lettera cercata non esiste, si stampa un messaggio di lettera assente.
Linee **4010-4160**. Appena trovato il gruppo di undici nominativi fra cui si trovano anche i nominativi inizianti con la lettera richiesta, si visualizzano con la routine presente tra le linee **4700** e la **4760**, gli undici nominativi appena trovati.
Alla linea **4030**, l'algoritmo attende il comando di posizionamento cursore. Se il tasto premuto è + il valore del cursore aumenta, se invece si preme — questo valore decresce. L'aumentare o decrescere del valore del cursore permette alla freccetta presente sullo schermo di scendere o salire, in tal modo si può far puntare la freccetta sul cognome desiderato. Se il movimento del cursore supera gli undici nominativi presenti nello schermo, verranno visualizzati i precedenti o postecedenti undici utenti.
Linee **3310-3500**. Premendo il tasto RETURN si passa alla fase di visualizzazione di tutti i campi. In questa fase oltre a vedere tutti i campi del record, sono offerte 4 possibilità. La prima, la più semplice, ritorna alla fase precedente ossia la visualizzazione degli undici nominativi, la seconda permette di telefonare all'utente appena prescelto, la terza di

modificare i campi, infine la quarta cancella il nominativo dell'array.
Linee **5030-5420**. Occupiamoci ora della parte che permette di telefonare. La prima mossa dell'algoritmo e quella di testare se realmente a fianco la voce TEL. esista realmente un valore eseguibile. Se la condizione di eseguibilità non esiste allora si resta nella fase della visualizzazione utente. Se il numero è eseguibile l'algoritmo separa in due variabili distinte il prefisso ed il numero telefonico. Solo a questo punto chiede se si deve eseguire anche il prefisso e in caso affermativo memorizza in un unica variabile sia numero telefonico che prefisso.
Fatte queste operazioni l'algoritmo attende che venga premuto un tasto per avviare l'esecuzione del numero. Il numero telefonico memorizzato in D$, dalla linea **5320** alla **5350**, viene suddiviso in singole cifre e memorizzate nella variabile A$. Dalla linea **5280** alla **5310** vengono generati le interruzioni pari al valore posto in A$. Nell'integrato 6526 (complex interface adapter) esistono due timer chiamati rispettivamente A e B.
Questi timer opportunamente programmati generano delle temporizzazioni ben precise atte a pilotare le porte di output o di contare impulsi provenienti dall'esterno. I registri che controllano le operazioni dell'integrato sono allocati nella locazione di memoria 56590 per il registro A mentre per il registro B la locazione interessata è la 56591. Controllando i bit di questi registri si possono controllare tutte le funzioni generate dal 6526.
Per generare le interruzioni si carica nel byte basso e in quello alto del timer B un valore da decremetare ogni ciclo macchina. In seguito avvio un ciclo pari al numero di interruzioni da generare, setto il registro di controllo B in modo da impostare un segnale in uscita sulla porta PB7, carico il byte alto del registro con cui il medesimo valore caricato sul registro B, così facendo decremento di una unità ogni ciclo macchina il valore del timer B. Quando il valore del timer B corrisponde allo zero, cambio il valore della tensione alla porta PB7 e diseccito il relé collegato ad essa. Genero ancora un altro ritardo di 60 mS con un ciclo di FOR ed eseguo nuovamente la routine di interruzione. Appena finita la cifra si genera un altro ritardo con un ciclo di FOR di 500 mS e si passa all'esecuzione della cifra successiva.
Se dopo aver eseguito il numero telefonico la linea risultasse occupata, allora il processo appena descritto può essere nuovamente ripetuto.
Se invece non si vuol più telefonare al corrispondente del numero prescelto si torna alla visualizzazione dell'utente precedentemente selezionato.
Linee **3540-3750**. Se invece di telefonare si vuole modificare qualche campo dell'utente prescelto è necessario premere il tasto M che serve proprio per avviare la routine di modifica. La prima operazione di questa routine è quella di copiare sul vettore P$ (x) l'intero record desiderato. Poi l'algoritmo chiede quale elemento si vuole modificare e attende il comando. A tal punto si possono modificare tutti gli elementi del vettore con le procedure solite di input che abbiamo già descritto in precedenza.
Linee **4210-4260**. Tornati alla visualizzazione utente rimane da analizzare l'ultima possibilità, ovvero la cancellazione totale del nominativo dell'array. Questa opzione si avvia digitando il tasto C. La digitazione di questo tasto causa l'assegnazione di un asterisco nel terzo campo del record da cancellare e lo spostamento, in alto di un posto, di tutti i puntatori ai record a partire dal successivo puntatore del record che si vuole cancellare.
Descritte tutte le opzioni della sezione ricerca non resta che accennare all'ultima opzione del menù principale, la FINE LAVORO. Questa opzione si occupa semplicemente di aggiornare il file dati se vi sono state modifiche.
Ultima raccomandazione; evitate di inserire il connettore a computer acceso. Potrebbero verificarsi spiacevoli sorprese.

```
1000 REM  **************************
1010 REM  *                        *
1020 REM  *  RUBRICA TELEFONICA    *
1030 REM  *                        *
1040 REM  *       AUTOMATICA       *
1050 REM  *                        *
1060 REM  *           DI           *
1070 REM  *                        *
1080 REM  *     ERNESTO SIDOTI     *
1090 REM  *                        *
1100 REM  **E*********************S**
1110 PRINT"[CLEAR][BIANCO]":MA=198:
     POKE 650,128:POKE 56579,128:PO
     KE 56577,0
1120 POKE 54296,15:POKE 53280,0:POK
     E 53281,0
1130 DIM  NO$(MA+1,6):DIM P$(6):DIM
      X(MA+1):GOSUB 4800
1140 O$="  ----------------------
     ------------"
```

```
1150 REM   ************************
1160 REM   *                      *
1170 REM   *        MENU'         *
1180 REM   *                      *
1190 REM   ************************
1200 PRINT"[CLEAR]":ZZ=0:FL=0:HH=0:
     H=0:GOSUB 1840
1210 REM   PRINTFRE(0)
1220 PRINT TAB(6)"[DOWN]AGENDA TELE
     FONICA AUTOMATICA"
1230 PRINT TAB(9)"[DOWN]DI ERNESTO
     SIDOTI (C)"
1240 PRINT"[DOWN]├─────────────────
     ─────────────────┤"
1250 PRINT TAB(2)"F5> MEMORIZZARE"
1260 PRINT TAB(2)"[DOWN]"
1270 PRINT TAB(2)"[DOWN]F3> RICERCA
     RE"
1280 PRINT TAB(2)"[DOWN]"
1290 PRINT TAB(2)"[DOWN]F8> FINE LA
     VORO"
1300 PRINT"[DOWN]├─────────────────
     ─────────────────┤"
1310 PRINT"[HOME][7 DOWN]":A=20
1320 PRINT TAB(A)" ╭──────────╮"
1330 PRINT TAB(A)" |          |"
1340 PRINT TAB(A)" ╰──╯‾[RVS] [RVOF
     F]‾‾‾[RVS] [RVOFF]‾╰──╯"
1350 PRINT TAB(A)"   ╭──────╮"
1360 PRINT TAB(A)"   | [RVS] 7 8 9
     [RVOFF] |"
1370 PRINT TAB(A)"   | [RVS] 4 5 6
     [RVOFF] |"
1380 PRINT TAB(A)"   | [RVS] 1 2 3
     [RVOFF] |"
1390 PRINT TAB(A)"   | [RVS] * 0 #
     [RVOFF] |"
1400 PRINT TAB(A)"   ╰      ╯" "
1410 PRINT TAB(A)"   ────────"
1420 REM   ************************
1421 REM   *                      *
1430 REM   *    SCELTA DI MENU'   *
1431 REM   *                      *
1440 REM   ************************
1450 :
1460 GET SC$:IF  SC$="" THEN 1460
1470 IF SC$=CHR$(134) THEN 3810
1480 IF SC$=CHR$(135) THEN GOSUB 26
     50:GOTO 1200
1490 IF SC$=CHR$(140) THEN GOSUB 54
     60
1500 GOSUB 2590:GOTO 1460
1510 REM   ************************
1520 REM   *                      *
1530 REM   *      MASCHERA        *
1540 REM   *                      *
1550 REM   ************************
1560 PRINT"[CLEAR]":Z=2:GOSUB 1840
1570 GOSUB 1750
1580 PRINT"[HOME][DOWN][4 RIGHT]AGE
     NDA TELEFONICA AUTOMATICA  (C)
     "
1590 PRINT TAB(Z)"[DOWN][RVS]NOME"
1600 PRINT TAB(Z)"[DOWN][RVS]COGNOM
     E"
1610 PRINT TAB(Z)"[DOWN][RVS]INDIRI
     ZZO"
1620 PRINT TAB(Z)"[DOWN][RVS]COMUNE
     DI"
1630 PRINT TAB(Z)"[DOWN][RVS]TELEFO
     NO"
1640 PRINT TAB(Z)"[DOWN][RVS]C.A.P.
     "
1650 PRINT TAB(Z)"[DOWN][RVS]NOTE[R
     VOFF]"
1660 PRINT"[DOWN]├─────────────────
     ─────────────────┤":RETURN
1670 REM  **** PULITURA MASCHERA **
     ***
1680 PRINT TAB(Z)"[HOME][3 DOWN][6
     RIGHT] ..............."
1690 PRINT TAB(Z)"[DOWN][7 RIGHT] .
     ...................."
1700 PRINT TAB(Z)"[DOWN][9 RIGHT] .
     ........................"
1710 PRINT TAB(Z)"[DOWN][9 RIGHT] .
     ........................"
1720 PRINT TAB(Z)"[DOWN][8 RIGHT] .
     ...................."
1730 PRINT TAB(Z)"[DOWN][6 RIGHT] .
     ...."
1740 PRINT TAB(Z)"[DOWN][4 RIGHT] .
     .............................
     "
1750 PRINT"[DOWN]├─────────────────
     ─────────────────┤":RETURN
1760 REM   ************************
1761 REM   *                      *
1770 REM   *   PULIZIA FINESTRA   *
1771 REM   *                      *
1780 REM   ************************
1790 PRINT"[HOME][18 DOWN][2 RIGHT]

     "
1800 PRINT"[2 RIGHT]
                         "
```

```
1810 PRINT"[2 RIGHT]
                              "
1820 PRINT"[2 RIGHT]
                              "
1830 PRINT"[2 RIGHT]
                              ":RETURN
1840 REM  ************************
1841 REM  *                      *
1850 REM  *      RIQUADRO        *
1851 REM  *                      *
1860 REM  ************************
1870 FOR I=55296 TO 55335:POKE I,1:
     POKE I+960,1:NEXT:FOR I=55336
     TO 56216 STEP 40:POKE I,1
1880 POKE I+39,1:NEXT
1890 POKE 1024,85:FOR I=1025 TO 106
     2:POKE I,67:NEXT:POKE 1063,73:
     FOR I=1103 TO 1983 STEP 40
1900 POKE I,66:NEXT:POKE 2023,75:FO
     R I=2022 TO 1985 STEP -1:POKE
     I,67:NEXT:POKE 1984,74
1910 FOR I=1944 TO 1064 STEP -40:PO
     KE I,66:NEXT:RETURN
1920 REM  ************************
1921 REM  *                      *
1930 REM  *       INPUT          *
1931 REM  *                      *
1940 REM  ************************
1950 X%=7:Y%=4:L%=15:T%=0:GOSUB 215
     0:GOSUB 2090
1960 RETURN
1970 X%=10:Y%=6:L%=20:T%=0:GOSUB 21
     50:GOSUB 2090
1980 RETURN
1990 X%=12:Y%=8:L%=26:T%=0:GOSUB 21
     50:GOSUB 2090
2000 RETURN
2010 X%=12:Y%=10:L%=26:T%=0:GOSUB 2
     150:GOSUB 2090
2020 RETURN
2030 X%=11:Y%=12:L%=24:T%=0:GOSUB 2
     150:GOSUB 2090
2040 RETURN
2050 X%=9:Y%=14:L%=5:T%=0:GOSUB 215
     0:GOSUB 2090
2060 RETURN
2070 X%=7:Y%=16:L%=31:T%=0:GOSUB 21
     50:GOSUB 2090
2080 RETURN
2090 IF  LEFT$(X$,1)=" "  THEN X$="
     "
2100 RETURN
2110 REM  ************************
2111 REM  *                      *
2120 REM  *  INPUT CONTROLLATO   *
2130 REM  *  VAR.=X%,Y%,L%,T%,X$ *
2131 REM  *                      *
2140 REM  ************************
2150 PRINT"[HOME]";:FOR AA=1 TO Y%-
     1:PRINT"[DOWN]";:NEXT:I%=0:IF
     Y%=1 THEN PRINT"[UP]";
2160 FOR AA=1 TO X%:PRINT"[RIGHT]";
     :NEXT:FOR AA=1 TO L%:PRINT" ";
     :NEXT:FOR AA=2 TO L%
2170 PRINT"[LEFT]";:NEXT:IF T%=1 TH
     EN C$=LEFT$(X$,1):IF C$="" OR
     VAL(C$)<>0 THEN C$="+"
2180 IF T%=1 THEN X$=C$+RIGHT$(STR$
     (VAL(X$)),LEN(STR$(VAL(X$)))-1
     ):I%=1
2190 IF T%=1 THEN PRINT"[LEFT]";
2200 PRINTX$;:IF T%=1 THEN PRINT"[R
     IGHT]";
2210 FOR AA=1 TO LEN(X$):PRINT"[LEF
     T]";:NEXT
2220 GET C$:IF C$<>"" AND FL=0 THEN
      2290
2230 IF C$="[HOME]" THEN 2120
2240 K%=PEEK(1024+(Y%-1)*40+X%+I%)
2250 IF K%>0 AND K%<127 THEN POKE 1
     024+(Y%-1)*40+X%+I%,K%+128:FL=
     1
2260 IF K%>127 AND K%<255 THEN POKE
      1024+(Y%-1)*40+X%+I%,K%-128:F
     L=0
2270 IF C$="" THEN 2220
2280 IF C$<>"" AND FL=1  THEN 2260
2290 IF C$="[HOME]" THEN 2120
2300 C%=ASC(C$):IF C%=145 THEN GOSU
     B 2560:GOTO 2220
2310 IF C%=13 THEN 2450
2320 IF T%=1 AND I%=0 THEN IF C%=43
      OR C%=45 THEN 2350
2330 IF T%=1 AND (C%=29 OR C%=157)
     THEN 2350
2340 IF T%=1 AND (C%<48 OR C%>57) T
     HEN GOSUB 2560:GOTO 2220
2350 IF C%=17 THEN GOSUB 2560:GOTO
     2220
2360 IF C%=157 AND I%<=0 THEN GOSUB
      2560:GOTO 2220
2370 IF C%=157 AND I%>0 THEN I%=I%-
     1:PRINT"[LEFT]";:GOTO 2220
2380 IF C%=29 AND I%>=L%-1 THEN GOS
     UB 2560:GOTO 2220
2390 IF C%=29 AND I%<L% THEN I%=I%+
```

```
     1:PRINT"[RIGHT]";:GOTO 2220
2400 IF T%=1 AND I%<>1 AND (C%=43 O
     R C%=45) THEN 2430
2410 IF T%=1 AND (C%<48 AND C%>57)
     THEN GOSUB 2560:GOTO 2220
2420 IF C%<32 OR C%>93 THEN GOSUB 2
     560:GOTO 2220
2430 IF I%>L%-1 THEN GOSUB 2560:GOT
     O 2220
2440 I%=I%+1:PRINTC$;:GOTO 2220
2450 X$="":IF T%=1 THEN FOR AA=0 TO
      L%-1:C=PEEK(1024+(Y%-1)*40+X%
     +AA):X$=X$+CHR$(C)
2460 IF T%=1 THEN NEXT:GOTO 2510
2470 REM
2480 FOR AA=0 TO L%-1:C=PEEK(1024+(
     Y%-1)*40+X%+AA)
2490 IF  C>27  THEN  C=C-64
2500 X$=X$+CHR$(C+64):NEXT
2510 FOR CB=L% TO 1  STEP -1
2520 IF MID$(X$,CB,1)<>" " THEN 254
     0
2530 NEXT:RETURN
2540 FOR CC=CB TO 1 STEP -1:V$=MID$
     (X$,CC,1)+V$:NEXT:X$=V$:V$=""
2550 RETURN
2560 REM  ************************
2561 REM  *.                     *
2570 REM  *        SOUND         *
2571 REM  *                      *
2580 REM  ************************
2590 POKE 54277,8:POKE 54278,17:POK
     E 54273,89:POKE 54272,15:POKE
     54274,0
2600 POKE 54275,9:POKE 54276,65
2610 FOR TT=1 TO 90:NEXT:POKE 54276
     ,0:RETURN
2620 REM  ************************
2621 REM  *                      *
2630 REM  *    MEMORIZZAZIONE    *
2631 REM  *                      *
2640 REM  ************************
2650 GOSUB 5570:GOSUB 1560:GOSUB 16
     80
2660 GOSUB 2680:GOSUB 2690:GOSUB 27
     00:GOSUB 2710:GOSUB 2720:GOSUB
      2730
2670 GOSUB 2740:GOTO 2750
2680 GOSUB 1950:P$(0)=X$:X$="":RETU
     RN
2690 GOSUB 1970:P$(1)=X$:X$="":RETU
     RN
2700 GOSUB 1990:P$(2)=X$:X$="":RETU
     RN
2710 GOSUB 2010:P$(3)=X$:X$="":RETU
     RN
2720 GOSUB 2030:P$(4)=X$:X$="":RETU
     RN
2730 GOSUB 2050:P$(5)=X$:X$="":RETU
     RN
2740 GOSUB 2070:P$(6)=X$:X$="":RETU
     RN
2750 PRINT"[HOME]"
2760 PRINT TAB(11)"[18 DOWN]VUOI CO
     RREGERE ?":PRINT:PRINT
2770 PRINT"[6 RIGHT](SI <RETURN>) (
     NO <SPACE>)"
2780 GET Z$:IF Z$="" THEN 2780
2790 IF Z$=CHR$(32) THEN 3000
2800 IF Z$=CHR$(13) THEN 2820
2810 IF Z$<>CHR$(32) OR Z$<>CHR$(13
     ) THEN 2780
2820 GOSUB 1790
2830 PRINT"[2 RIGHT][5 UP]COSA VUOI
      CORREGERE ?":PRINT
2840 PRINT"[2 RIGHT]1)NOME  2)COGNO
     ME  3)INDIRIZZO":PRINT
2850 PRINT"[2 RIGHT]4)TELEFONO  5)C
     OMUNE  6)C.A.P  7)NOTE
2860 GET J$:IF J$="" THEN 2860
2870 IF J$="1" THEN GOSUB 2680:GOTO
      2940
2880 IF J$="2" THEN GOSUB 2690:GOTO
      2940
2890 IF J$="3" THEN GOSUB 2700:GOTO
      2940
2900 IF J$="4" THEN GOSUB 2720:GOTO
      2940
2910 IF J$="5" THEN GOSUB 2710:GOTO
      2940
2920 IF J$="6" THEN GOSUB 2730:GOTO
      2940
2930 IF J$="7" THEN GOSUB 2740:GOTO
      2940
2940 GOSUB 1790:GOTO 2750
2950 REM  ************************
2960 REM  *                      *
2970 REM  *  RINNOVO PUNTATORI   *
2980 REM  *                      *
2990 REM  ************************
3000 IF  X(1)=0  THEN X(1)=1:FOR A=
     0 TO 6:NO$(1,A)=P$(A):NEXT:RET
     URN
3010 FOR D=0 TO 6:NO$(PO,D)=P$(D):N
     EXT
3020 C1=PO:CL=CL-1
```

```
3030 FOR A=1  TO  CL
3040 IF NO$(X(A),1)>P$(1) AND NO$(X
     (A),1)<>P$(1)  THEN  GOTO 3090
3050 NEXT
3060 IF  PO=1  THEN  X(CL)=PO:RETUR
     N
3070 IF  CL< MA  THEN  X(CL+1)=PO:R
     ETURN
3080 IF  CL=MA  THEN  X(CL)=PO:RETU
     RN
3090 FOR B=CL  TO  A  STEP -1
3100 X(B+1)=X(B)
3110 NEXT
3120 X(A)=PO
3130 RETURN
3140 REM  ************************
3150 REM  *                      *
3160 REM  *  RICERCA FINE ARRAY  *
3170 REM  *                      *
3180 REM  ************************
3190 FOR  CL=1  TO  MA
3200 IF  NO$(CL,1)="" THEN RETURN
3210 NEXT
3220 IF FL=1  THEN  FL=0:RETURN
3230 PRINT"[CLEAR]":PRINT TAB(14)"[
     7 DOWN]AGENDA PIENA"
3240 FOR A=1 TO 5000:NEXT
3250 GOTO 1200
3260 REM  ************************
3270 REM  *                      *
3280 REM  * STAMPA UTENTE SCELTO *
3290 REM  *                      *
3300 REM  ************************
3310 IF NO$(X(CUR-1),1)="" THEN IT=
     I2:CU=C2:GOTO 4020
3320 PRINT"[CLEAR]"
3330 GOSUB 1560:FOR A=0  TO  6:P$(A
     )=NO$(X(CUR-1),A):NEXT
3340 PRINT"[HOME][DOWN]"
3350 PRINT  TAB(7)"[DOWN]"P$(0)
3360 PRINT  TAB(10)"[DOWN]"P$(1)
3370 PRINT  TAB(12)"[DOWN]"P$(2)
3380 PRINT  TAB(12)"[DOWN]"P$(3)
3390 PRINT  TAB(11)"[DOWN]"P$(4)
3400 PRINT  TAB(9)"[DOWN]"P$(5)
3410 PRINT  TAB(7)"[DOWN]"P$(6)
3420 GOSUB 1790:PP=15:GOSUB 4300
3430 PRINT  TAB(5)"<T> TELEFONARE  <
     M> MODIFICARE"
3440 PRINT TAB(7)"[DOWN]<C> CANCELL
     ARE  <←> ESCE"
3450 GET SC$:IF SC$="" THEN 3450
3460 IF SC$="←" THEN CU=IT+1:GOTO 4
     020
3470 IF  SC$="T"  THEN 5030
3480 IF  SC$="M"  THEN 3540
3490 IF  SC$="C"  THEN 4210
3500 GOSUB 2590:GOTO 3450
3510 REM  ************************
3511 REM  *                      *
3520 REM  *     CORREZIONE       *
3521 REM  *                      *
3530 REM  ************************
3540 GOSUB 1790:PP=14:GOSUB 4300
3550 PRINT  TAB(10)"[DOWN][RVS]COSA
     VUOI CORREGGERE"
3560 PRINT TAB(5)"[DOWN]1)NOME   2)I
     NDIRIZZO  3)COMUNE"
3570 PRINT  TAB(8)"[DOWN]4)TEL    5)C
     AP   6)NOTE"
3580 GET  SX$:IF SX$="" THEN 3580
3590 IF ASC(SX$)<49  OR  ASC(SX$)>5
     5  THEN 3580
3600 IF SX$="1" THEN  GOSUB 1950:P$
     (0)=X$:X$=""
3610 REM  IFSX$="2"THEN GOSUB710:P$
     (1)=X$:X$=""
3620 IF SX$="2" THEN  GOSUB 1990:P$
     (2)=X$:X$=""
3630 IF SX$="3" THEN  GOSUB 2010:P$
     (3)=X$:X$=""
3640 IF SX$="4" THEN  GOSUB 2030:P$
     (4)=X$:X$=""
3650 IF SX$="5" THEN  GOSUB 2050:P$
     (5)=X$:X$=""
3660 IF SX$="6" THEN  GOSUB 2070:P$
     (6)=X$:X$=""
3670 GOSUB  1790:PP=15:GOSUB 4300
3680 PRINT TAB(13)"TUTTO ESATTO  ?"
3690 PRINT TAB(9)"[DOWN]SI <RETURN>
      NO<SPACE>"
3700 GET SX$:IF (SX$="") OR ((SX$<>
     CHR$(13)) AND (SX$<>CHR$(32)))
      THEN 3700
3710 IF SX$=CHR$(13)  THEN  GOTO 37
     30
3720 GOTO  3540
3730 FOR A=0 TO 6
3740 NO$(X(CUR-1),A)=P$(A)
3750 NEXT:GOTO 3420
3760 REM  ************************
3770 REM  *                      *
3780 REM  *       RICERCA        *
3790 REM  *                      *
3800 REM  ************************
3810 PRINT"[CLEAR]":GOSUB 1870:PP=1
```

```
 0:GOSUB 4300:PRINT TAB(12)"QUA
 LE LETTERA ?"
3820 GET SC$:IF SC$="" OR SC$<CHR
 $(65) OR SC$>CHR$(90) THEN
 GOTO 3820
3830 PRINT TAB(18)"[DOWN]┌─┐":PRINT
 TAB(18)"|"SC$"|":PRINT TAB(18
 )"└─┘"
3840 FL=1:GOSUB 3190:CL=CL-1
3850 FOR A=1 TO CL STEP 11
3860 FOR B=A TO A+10
3870 IF LEFT$(NO$(X(B),1),1)=SC$ T
 HEN IT=A:CUR=IT+1:I2=IT:C2=CU:
 GOTO 4020
3880 NEXT
3890 NEXT
3900 PRINT TAB(4)"[DOWN]LETTERA NON
 PRESENTE IN ARCHIVIO"
3910 PRINT TAB(8)"[3 DOWN]TORNO AL
 MENU' ? (S/N)"
3920 GET SC$:IF SC$="" THEN 3920
3930 IF SC$="N" THEN 3810
3940 IF SC$="S" THEN 1200
3950 GOSUB 2590:GOTO 3920
3960 REM ************************
3970 REM *                      *
3980 REM * SPOSTAMENTO CURSORE  *
3990 REM *                      *
4000 REM ************************
4010 PRINT"[CLEAR]":CUR=2:IT=1
4020 GOSUB 4700:PRINT"[HOME][DOWN]-
 ->"
4030 GET S$:CUR=CUR+(S$="-")-(S$="+
 ")
4040 IF S$="" THEN GOTO 4030
4050 IF S$=CHR$(13) THEN 3310
4060 IF S$="←" THEN 1200
4070 IF CUR-1>MA THEN 4010
4080 IF CUR-1<1 THEN IT=188:CU=189
 :GOTO 4020
4090 IF S$="-" THEN J=J-1
4100 IF CUR>IT+11 THEN IT=IT+11:GO
 SUB 4700:PRINT"[HOME][DOWN]-->
 ":GOTO 4140
4110 IF CUR=<IT THEN IT=IT-11:GOSU
 B 4700:PP=19:GOSUB 4300:PRINT"
 -->":GOTO 4140
4120 IF S$="+" THEN PRINT"[2 UP]":P
 RINT"   ":PRINT"   ":PRINT"-->
 "
4130 IF S$="-" THEN PRINT"[3 UP]-->
 ":PRINT"   ":PRINT"   ":PRINT"
 [3 UP]"
4140 GOTO 4030
4150 END
4160 PRINT TAB(2)"[UP]"NO$(J+1,0)
4170 RETURN
4180 REM ************************
4181 REM *                      *
4190 REM *   CANCELLA UTENTE    *
4191 REM *                      *
4200 REM ************************
4210 IN=CU-1:NO$(X(IN),2)="*"
4220 FOR A=IN TO CL
4230 IF A=MA THEN X(A)=0:GOTO 426
 0
4240 X(A)=X(A+1)
4250 NEXT
4260 IT=I2:CU=C2:GOTO 4020
4270 REM ************************
4271 REM *                      *
4280 REM *  POSIZIONE  STAMPA   *
4281 REM *                      *
4290 REM ************************
4300 PRINT"[HOME]":FOR PS=0 TO PP:
 PRINT:NEXT:RETURN
4310 REM ************************
4311 REM *                      *
4320 REM *  SCRITTURA  DISCO    *
4321 REM *                      *
4330 REM ************************
4340 OPEN 15,8,15,"S:DATA":CLOSE 15
4350 OPEN 2,8,2,"0:DATA,S,W"
4360 FL=1:GOSUB 3190
4370 FOR M=1 TO CL-1
4380 FOR P=0 TO 6
4390 PRINT#2,NO$(M,P)CHR$(44)
4400 NEXT
4401 W$=STR$(X(M))
4410 PRINT#2,W$CHR$(44)
4420 NEXT
4421 PRINT#2,"←"CHR$(44)
4430 CLOSE 2:CLOSE 15
4440 RETURN
4450 REM ************************
4451 REM *                      *
4460 REM *    LETTURA DISCO     *
4461 REM *                      *
4470 REM ************************
4480 OPEN 2,8,2,"0:DATA,S,R"
4490 GOSUB 4620
4500 M=1
4510 FOR P=0 TO 6
4520 INPUT#2,W$
4521 IF W$="←" THEN 4570
4522 NO$(M,P)=W$
```

```
4530 NEXT
4540 INPUT#2,W$
4541 X(M)=VAL(W$)
4560 M=M+1:GOTO 4510
4570 CLOSE 2:CLOSE 15
4580 RETURN
4590 REM  ************************
4591 REM  *                      *
4600 REM  *  CONTROLLO  ERRORE   *
4601 REM  *                      *
4610 REM  ************************
4620 OPEN 15,8,15
4630 INPUT#15,EN,EM$,ET,ES
4640 IF  EN=0  THEN  RETURN
4650 PRINT"[CLEAR]":PRINT TAB(12)"[
     3 DOWN][RVS] ERRORE SU DISCO [
     RVOFF][3 DOWN]"
4660 PRINT:PRINTEM$:PRINT
4670 PRINT:PRINT"ERRORE TIPO",EN
4680 PRINT:PRINT"TRACCIA     ",ET
4690 PRINT:PRINT"SETTORE     ",ES:PR
     INT:CLOSE 15:END
4700 IF  CUR=MA-1 THEN CUR=189:IT=1
     88:GOTO 4020
4710 IF IT<1 THEN IT=1:GOTO 4010
4720 PRINT"[CLEAR]":FOR J=IT TO IT+
     10:PRINT TAB(4)NO$(X(J),1) TAB
     (24)NO$(X(J),0):PRINTO$:NEXT
4730 PG=J/11
4740 GU=CUR
4750 PRINT TAB(16)"PAGINA   ";"[3 L
     EFT]"INT(PG)
4760 PRINT TAB(4)"[RVS]<+/->CUR  <←
     >MENU'  <RETURN>SCELTA[RVOFF][
     UP]":RETURN
4770 REM  ************************
4771 REM  *                      *
4780 REM  *       INIZIO         *
4781 REM  *                      *
4790 REM  ************************
4800 PRINT"[CLEAR]":GOSUB  1870
4810 PP=7:GOSUB 4300
4820 PRINT TAB(15)"COSA USI ?"
4830 PRINT TAB(12)"[2 DOWN][RVS]D[R
     VOFF]ISCO O [RVS]C[RVOFF]ASSET
     TA"
4840 GET SC$:IF SC$="" THEN 4840
4850 IF SC$="D"  THEN DD=0:GOTO 488
     0
4860 IF SC$="C"  THEN DD=1:GOTO 488
     0
4870 GOSUB 2590:GOTO 4840
4880 PRINT"[CLEAR]":GOSUB  1870
4890 PP=7:GOSUB 4300
4900 PRINT TAB(8)"[RVS]
                         "
4910 PRINT TAB(8)"[RVS]
                         "
4920 PRINT TAB(8)"[RVS] INSERIRE SU
     PPORTO DATI "
4930 PRINT TAB(8)"[RVS]
                         "
4940 PRINT TAB(8)"[4 DOWN]NUOVO ARC
     HIVIO ?   (S/N)"
4950 GET SC$:IF SC$="" THEN 4950
4960 IF SC$="S"  THEN K%=1:RETURN
4970 IF SC$="N"  AND  DD=0  THEN GO
     SUB 4480:RETURN
4980 IF SC$="N"  AND  DD=1  THEN GO
     SUB 5830:RETURN
4990 GOSUB 2590:GOTO 4950
5000 REM  ************************
5001 REM  *                      *
5010 REM  *     TELEFONARE       *
5011 REM  *                      *
5020 REM  ************************
5030 H$=LEFT$(NO$(X(CUR-1),4),1)
5040 IF  H$<CHR$(48)  OR  H$>CHR$(5
     7) THEN GOSUB 2590:GOTO 3420
5050 PRINT"[CLEAR]":GOSUB 1870
5060 FOR A=1  TO  LEN(P$(4))
5070 IF MID$(P$(4),A,1)="/" THEN  O
     =1:GOTO  5100
5080 M$=M$+MID$(P$(4),A,1)
5090 NEXT:IF O=1 THEN  N$=M$:M$="":
     O=0:GOTO 5120
5100 FOR B=A+1  TO  LEN(P$(4))
5110 N$=N$+MID$(P$(4),B,1):NEXT:O=0
5120 PRINT TAB(8)"[3 DOWN][RVS]
                         "
5130 PRINT TAB(8)"[RVS] ALZARE IL M
     ICROTELEFONO "
5140 PRINT TAB(8)"[RVS]
                    [RVOFF]"
5150 PRINT TAB(5)"[2 DOWN]NUMERO DA
      ESEGUIRE "P$(4)
5160 PRINT TAB(4)"[3 DOWN]ESEGUO AN
     CHE IL PREFISSO  ?  (S/N)"
5170 GET SC$:IF SC$="" THEN 5170
5180 IF  SC$="S" THEN  D$=M$+N$:GOT
     O 5210
5190 IF  SC$="N" THEN  D$=N$:GOTO 5
     210
5200 GOSUB 2590:GOTO 5170
5210 POKE 56579,255:POKE 56577,128:
     FOR A=0 TO 2000:NEXT:POKE 5657
```

```
     7,0
5220 PP=12:GOSUB 4300:PRINT TAB(5)"
     [2 DOWN][RVOFF]PREMERE UN TAST
     O A LINEA LIBERA":FOR T=1 TO 1
     00:NEXT
5230 GET  VV$:IF VV$<>"" THEN 5320
5240 PP=12:GOSUB 4300:PRINT TAB(5)"
     [2 DOWN][RVS]PREMERE UN TASTO
     A LINEA LIBERA":FOR T=1 TO 100
     :NEXT
5250 GOTO 5220
5260 GOTO 5320
5270 POKE 56590,1
5280 POKE 56582,0
5290 POKE 56583,175
5300 FOR S=1 TO VAL(A$):POKE 56591,
     71:POKE 56581,175:POKE 56591,6
     9:FOR I=1 TO 57:NEXTI:NEXT
5310 FOR X=1 TO 500:NEXT:RETURN
5320 FOR A=1 TO LEN(D$)
5330 A$=MID$(D$,A,1):IF A$="0" THEN
      A$="10"
5340 GOSUB 5270
5350 NEXTA
5360 PP=18:GOSUB 4300:PRINT TAB(11)
     "                             "
5370 PRINT TAB(11)"[UP][RVS] RITENT
     O ? <S/N> "
5380 GET S$:IF S$="" THEN 5380
5390 IF S$="S" THEN PP=18:GOSUB 430
     0:PRINT TAB(11)"
                ":GOTO 5420
5400 IF S$="N" THEN A$="":M$="":N$=
     "":GOTO 3310
5410 IF S$<>"N" OR  S$<>"S"  THEN 5
     380
5420 PP=12:GOSUB 4300:PRINT TAB(5)"
     [2 DOWN]
                ":GOTO 5210
5430 REM  ************************
5431 REM  *                      *
5440 REM  *         FINE         *
5441 REM  *                      *
5450 REM  ************************
5460 PRINT"[CLEAR]":GOSUB 1870:PP=1
     0:GOSUB 4300
5470 PRINT TAB(7)"HAI FATTO MODIFIC
     HE ?  <S/N>":PRINT"[2 DOWN][7
     RIGHT]";
5480 GET SC$:IF SC$="" THEN 5480
5490 IF SC$="S"  AND  DD=0 THEN GOS
     UB 4340:GOTO 5530
5500 IF SC$="S"  AND  DD=1 THEN GOS
     UB 5700:GOTO 5530
5510 IF SC$="N"  THEN 5530
5520 GOSUB 2590:GOTO 5480
5530 PRINT TAB(7)"[3 UP]          ARR
     IVEDERCI           ":END
5540 REM  ************************
5541 REM  *                      *
5550 REM  *     CERCA POSTO      *
5551 REM  *                      *
5560 REM  ************************
5570 FL=1:GOSUB 3190
5580 FOR A=1  TO  CL
5590 IF NO$(A,2)="*" THEN GOTO 5640
5600 IF CL=MA THEN 3230
5610 NEXT
5620 IF  CL< MA  THEN  PO=CL:RETURN
5630 GOTO 3230
5640 PO=A
5650 RETURN
5670 REM  ************************
5671 REM  *                      *
5680 REM  *    SCRIVE SU TAPE    *
5681 REM  *                      *
5690 REM  ************************
5700 OPEN 1,1,1,"DATA"
5710 FL=1:GOSUB 3190
5720 FOR M=1 TO CL-1
5730 FOR  P=0  TO  6
5740 PRINT#1,NO$(M,P)CHR$(44)
5750 NEXT
5751 W$=STR$(X(M))
5760 PRINT#1,W$CHR$(44)
5770 NEXT
5771 PRINT#1,"←"CHR$(44)
5780 CLOSE 1
5790 RETURN
5800 REM  ************************
5801 REM  *                      *
5810 REM  *    LEGGE DA TAPE     *
5811 REM  *                      *
5820 REM  ************************
5830 OPEN 1,1,0,"DATA"
5840 M=1
5850 FOR  P=0  TO  6
5860 INPUT#1,W$
5861 IF W$="←" THEN  4570
5862 NO$(M,P)=W$
5870 NEXT
5880 INPUT#1,W$
5881 X(M)=VAL(W$)
5900 M=M+1:GOTO 5850
5910 CLOSE 1
5920 PRINT"[CLEAR]":RETURN
```

LE OSCILLAZIONI

di **Mauro.Massetti**

(il problema del pendolo)

In fisica, uno dei fenomeni che riveste notevole interesse e curiosità è senza dubbio quello del moto armonico. In molti musei di scienza e tecnica, in particolare quello di Milano, si trova una intera stanza ove è situato un pendolo semplice con una corda che ha una lunghezza di oltre dieci metri ed una "boccia" di molti chilogrammi.

Il pendolo semplice: l'esempio più comune di moto armonico semplice. Si definisce "pendolo semplice" un qualsivoglia corpo di massa M sospeso per un filo di lunghezza definita L. In questo caso si ipotizza che lo spostamento X rispetto alla posizione di riposo (quella posizione in cui il pendolo da noi definito resta immobile in assenza di sollecitazioni perturbanti esterne) sia molto piccolo rispetto a L. In natura però non si trovano esempi di moti armonici semplici in senso assoluto: esistono un'infinità di "perturbazioni" che influiscono sul moto stesso.

Queste "perturbazioni" danno comunemente origine a smorzamenti che portano all'esaurirsi del moto considerato.

Uno degli esempi di causa di smorzamento più comuni è legato alla densità del mezzo nel quale avviene il moto (tipico l'aria). Questo particolare tipo di smorzamento è difficile da calcolare con precisione, perché funzione di parecchi parametri: la rugosità superficiale del corpo, la sua forma e quindi il suo coefficiente di penetrazione (il famoso CX automobilistico) ecc.

Più semplice da calcolare risulta invece lo smorzamento derivante da eventuali urti con altri pendoli semplici. A tal proposito si consideri il gioco, se così lo si può chiamare, delle cinque sfere sospese al filo che battono l'una contro l'altra in maniera tale che quelle di estrema destra e di estrema sinistra si staccano alternativamente dalle altre.

In questo caso lo smorzamento è dato dal grado di elasticità delle sfere stesse: più alto è il grado di elasticità, più basso è il fattore di smorzamento e quindi più a lungo il sistema di pendoli si manterrà in movimento.

Il programma che vi propongo è relativo al fenomeno sopra descritto. La sua semplicità non pone particolari problemi di comprensione e di utilizzo: una volta visualizzate le cinque sfere sullo schermo dopo il "RUN" iniziale è sufficiente "caricare" il sistema premendo più volte il tasto CRSR (quello all'estrema destra della vostra tastiera) per portare la sfera di destra alla posizione dalla quale si desidera inizi il movimento e quindi premere un tasto qualsiasi. A questo punto si attende che il sistema si riporti alle condizioni di equilibrio (l'immobilità). Per terminare, battere nuovamente un tasto qualsiasi. I valori del fattore di smorzamento FS sono compresi nell'intorno 0<=FS <=1; per FS=0 non vi è smorzamento ed il moto continua all'infinito (è quasi il caso di corpi perfettamente elastici) mentre per FS= il moto si esaurisce al primo impatto.

Per la realizzazione della parte grafica si è ricorsi all'utilizzo del "SIMONS' BASIC", un tool (strumento addizionale) avanzato disponibile per il Commodore 64 che mette a disposizione del programmatore ben 114 comandi addizionali rispetto al BASIC standard.

Il Simon's BASIC è nato dalla fantasia, dalla tenacia e dalla passione per l'informatica di David Simons, un ragazzino allora tredicenne, che utilizzando il Commodore appena regalatogli dal padre volle dotarlo di un set di funzioni addizionali che portassero le capacità della macchina a sua disposizione a competere con quelle di altre per le quali erano disponibili versioni BASIC più potenti. Questo BASIC risulta essere quindi un "collage" di comandi e funzioni appartenenti sia a Home che a Personal Computers.

Nel caso specifico sono state utilizzate solo alcune fra le funzioni disponibili per la grafica e cioè:

- **HIRES**, che serve ad inizializzare in modo grafico ad alta risoluzione lo schermo.
- **LINE**, che traccia una retta fra due punti dati.
- **CIRCLE**, che traccia un cerchio di raggio e centro definiti.
- **PAINT**, che riempie una superficie chiusa con un colore definito.
- **PAUSE**, che ferma l'elaborazione per un numero di secondi predeterminato.
- **DESIGN**, che permette di creare con facilità uno sprite.
- **MOB SET**, che alloca nella memoria del C641 o sprite appena creato.
- **MMOB**, che richiama uno sprite precedentemente creato, lo attiva e lo visualizza sullo schermo nella posizione desiderata.
- **RLOCMOB**, che sposta nella nuova posizione voluta uno sprite precedentemente attivato.
- **VOL**, che sceglie il volume del suono.
- **WAVE**, che sceglie la forma d'onda.
- **ENVELOPE**, per definire le fasi del suono.
- **MUSIC**, che permette di comporre o creare l'effetto sonoro.
- **PLAY**, per suonare quanto precedentemente composto.

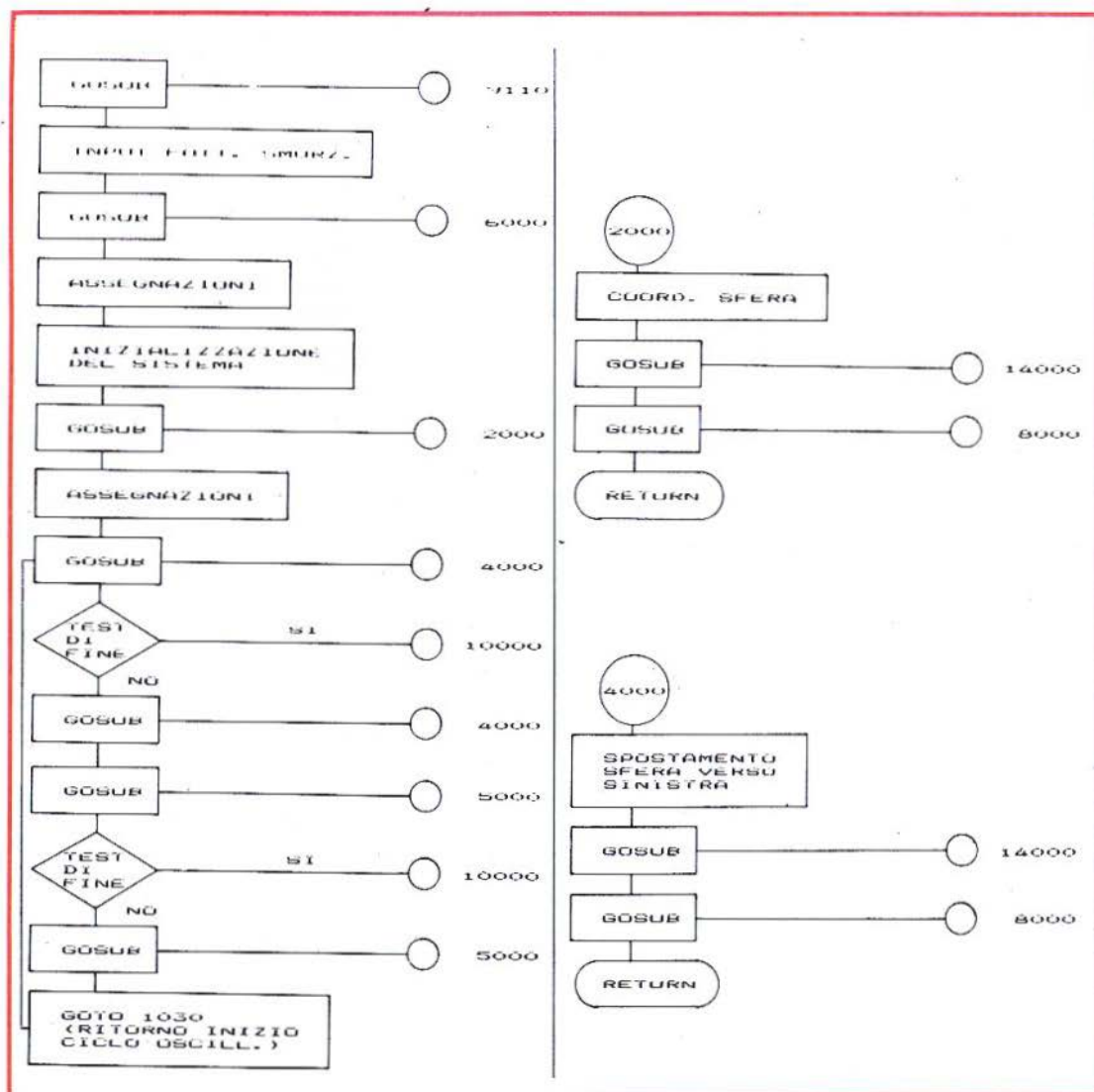
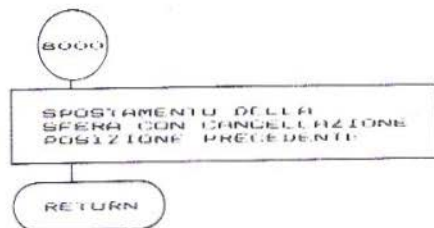
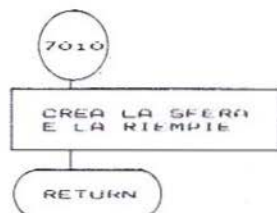
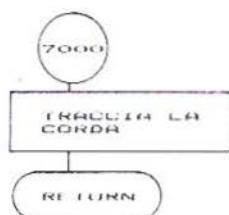
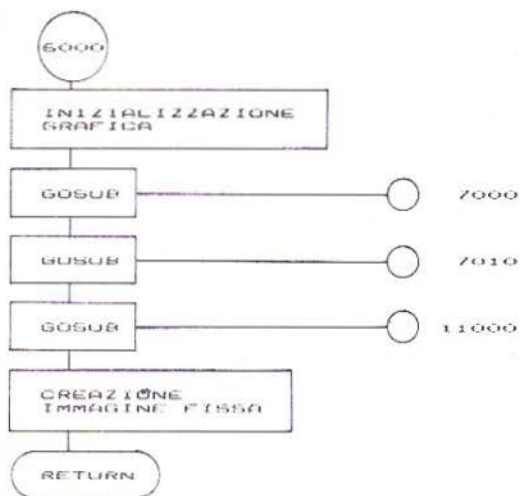
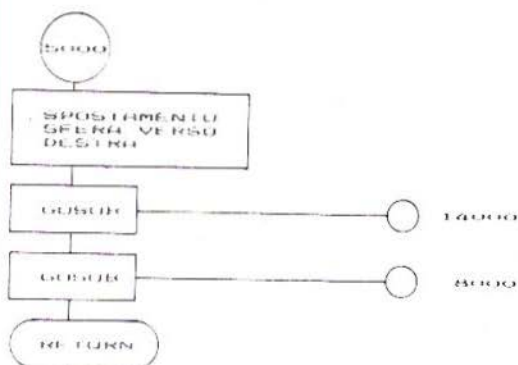
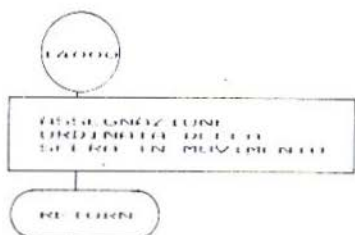
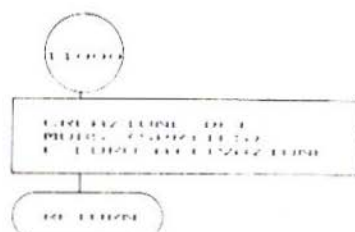
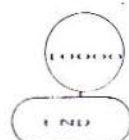
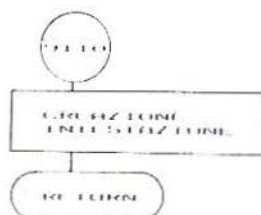
Passiamo ora ad analizzare più in dettaglio il programma propostovi. Dopo un rinvio alla subroutine di presentazione (linee **9100-9207**) viene richiesto il solo parametro definibile dall'utente necessario per far eseguire il programma: il fattore di smorzamento (ricordate che deve essere compreso fra zero e uno).

Seguono un nuovo rinvio alla routine grafica di creazione (linee **5900-7010**) assegnazioni di posizione e al

routine di "caricamento" del sistema di pendoli (linee **1011-1020**). A questo punto è allocata la routine di ritorno del pendolo costituito dalla sfera di destra ed il test di fine oscillazione (linee **1021-1060**); segue quindi la routine di allontanamento del pendolo costituito dalla sfera di sinistra (linee **1061-1070**) e le routines di movimento inverso (linee **1071-1120** e **1121-1130**) con il ritorno alla ripetizione del ciclo.

Le linee **1900-2010** sono relative alla subroutine richiamata nella linea **1019**; le linee **3900-4020** e **4900-5020** sono relative alla subroutines di spostamento verso sinistra e verso destra rispettivamente dei due pendoli in movimento. Alle linee **9900-10010** è allocato il test di fine programma. La creazione degli sprites è realizzata con le istruzioni che partono dalla linea 10950 alla 12230. È infine da notare che solo le due sfere ai lati si spostano in quanto il movimento si propaga istantaneamente attraverso il sistema di pendoli.

Provate dunque a variare il fattore di smorzamento eseguendo più prove onde poter confrontare le risposte del sistema utilizzato e non solo buon divertimento, ma ...buono studio della fisica!

5000
SPOSTAMENTO SFERA VERSO DESTRA
GOSUB
14000
GOSUB
8000
RETURN
6000
INIZIALIZZAZIONE GRAFICA
GOSUB
7000
GOSUB
7010
GOSUB
11000
CREAZIONE IMMAGINE FISSA
RETURN
7000
TRACCIA LA CORDA
RETURN
7010
CREA LA SFERA E LA RIEMPIE
RETURN
8000
SPOSTAMENTO DELLA SFERA CON CANCELLAZIONE POSIZIONE PRECEDENTE
RETURN

```
100 REM ********************
110 REM * IN SIMON'S BASIC *
120 REM ********************
130 REM
1000 REM**********************
1001 REM*  IL PROBLEMA DEL   *
1002 REM*      PENDOLO:      *
1003 REM*  LE OSCILLAZIONI   *
1004 REM*                    *
1005 REM* DI MAURO MASSETTI  *
1006 REM**********************
1007 REM* INPUT E ASSEGNAZ.  *
1008 REM**********************
1009 GOSUB9110:PRINT"▒▒":INPUT"FATT. SMORZ.";FS:GOSUB6000
1010 A=200:LC=140:K=200:K1=120:X=K:Y=LC
1011 REM*********************
1012 REM* INIZIALIZZAZIONE *
1013 REM*   DEL SISTEMA E   *
1014 REM*   ASSEGNAZIONI    *
1017 REM*********************
1018 GETS$:IFS$=""THEN1018
1019 IFS$="▒"THENIFX<=305THENGOSUB2000:GOTO1018
1020 SM=1-FS:XD=X:LS=K:A=200:XP=X:YP=Y
1021 REM***********************
```

```
11099 @BBBBBBBBBBBBBBBBBBBB....
11100 @BBBBBBBBBBBBBBBBBBBB....
11120 @BBBBBBBBBBBBBBBBBBBB....
11130 @.BBBBBBBBBBBBBBBBB......
11140 @.BBBBBBBBBBBBBBBBB......
11150 @..BBBBBBBBBBBBBBB.......
11160 @...BBBBBBBBBBBBB........
11162 @.....BBBBBBBBB..........
11164 @........................
11170 @........................
11180 @........................
11190 @........................
11200 @........................
11210 @........................
11220 [illegible] 0,M1,0,0,0
12000 M2=55:[illegible] 0,M2*64+49152
12010 @.....BBBBBBBBB..........
12020 @...BBBBBBBBBBBBB........
12040 @..BBBBBBBBBBBBBBB.......
12050 @.BBBBBBBBBBBBBBBBB......
12060 @.BBBBBBBBBBBBBBBBB......
12079 @BBBBBBBBBBBBBBBBBBBB....
12080 @BBBBBBBBBBBBBBBBBBBB....
12099 @BBBBBBBBBBBBBBBBBBBB....
12100 @BBBBBBBBBBBBBBBBBBBB....
12120 @BBBBBBBBBBBBBBBBBBBB....
12130 @.BBBBBBBBBBBBBBBBB......
12140 @.BBBBBBBBBBBBBBBBB......
12150 @..BBBBBBBBBBBBBBB.......
12160 @...BBBBBBBBBBBBB........
12162 @.....BBBBBBBBB..........
12164 @........................
12170 @........................
12180 @........................
12190 @........................
12200 @........................
12210 @........................
12220 [illegible] 1,M2,0,0,0
12230 RETURN
13900 REM************************
13901 REM* SUBR[illegible]INE POSIZIONE *
13902 REM* ORDINATE DELLO SPRITE*
13903 REM************************
14000 XX=ABS(A-X)
14010 IFXX<=20THENY=LC:GOTO14060
14020 IFXX<=30THENY=LC-1:GOTO14060
14030 IFXX<=60THENY=LC-(XX-30)/5-1:GOTO14060
14040 IFXX<=95THENY=LC-(XX-60)/2.5-7:GOTO14060
14050 Y=LC-(XX-95)/5*3-21
14060 RETURN

READY.
```

I programmi costituiti da una sola riga possono essere utility, "inutility" od altro. Le idee espresse sono e saranno solamente spigolature.
La grafica, la gestione dei video, la gestione delle stringhe: sono gli argomenti più semplici da sviluppare. Questo fatto non deve limitare la fantasia del lettore che invia ed invierà le 1R, ed a maggior ragione anche di chi le utilizzerà.

1 RIGA

La seconda puntata di "una riga" vede ancora gli ottimi lavori dei lettori. Queste 1R saranno remunerate con un libro scelto fra quelli proposti.
La disponabilità di premi è purtroppo limitata. Quindi, quando detta disponibilità sarà esaurita l'unica gratificazione sarà quella di vedersi pubblicati i propri lavori. Spero che questo fatto non sia un freno alla vostra fantasia.

In questo numero, troviamo in particolare tre 1R che possono essere un validissimo aiuto per coloro che sono in grado di eseguire i controlli di deflessione di un televisore. Se le capacità elettroniche non fossero così elevate, le 1R in questione possono essere utili per la comprensione della gestione del video.

Prima di esaminare le "una riga" di questa puntata, quali potranno essere gli argomenti su cui lavorare? Come eseguire l'hardcopy video? Come fare una cornice sullo schermo? Come fare una spirale quadrata sul video?

Alla prossima puntata e ...... buon lavoro

(Gloriano Rossi)

32

Colmaschermo. Una delle prime routine che si digitano quando si compera un Computer è quella che consente di riempire completamente il video con un medesimo carattere. Questo breve programma consente di scegliere di volta in volta l'elemento che si vuole utilizzare per compiere tale operazione.

(Roberto Morassi)

```
1 GET A$:PRINT"[HOME]"A$:FOR X=0 TO
  999:POKE 1025+X,PEEK(1024):NEXT:
  GOTO 1
```

33

Scacchiera. Non abbiamo l'ambizione di proporre in questa sede un gioco degli scacchi, anche perchè sarà difficile fare un programma di scacchi in una sola riga per il vostro computer Commodore, ma l'effetto visivo che si ottiene digitando questa routine è senza dubbio simile al teatro di battaglia di tante sfide scacchistiche.

(Roberto Morassi)

```
1 PRINT"[RVS] ";:POKE 646,(A  AND 1
  )+((TI/3E3) AND 15):B=B+1+40*(B=3
  9):A=A+1+(B=39):GOTO 1
```

34

Barre orizzontali. Un ottimo modo per sintonizzare il nostro televisore con il compuetr è avere stampato sullo schermo delle linee ravvicinate (orizzontali e verticali): quando saranno ben distinte la sintonizzazione sarà perfetta.

(Roberto Morassi)

```
1 POKE 646,N:FOR X=0 TO 39:PRINT"[R
  VS] ";:NEXT:N=N+1+16*(N=15):GOTO
  1
```

35

Labirinto. Con questa routine è possibile creare un labirinto a due dimensioni avvalendosi semplicemente dei caratteri messi a disposizione dal vostro computer COMMODORE (in questo caso C64).

(Roberto Morassi)

```
1 GET A$:PRINT"[HOME]"A
  $;:A=RND(1)*999+1025:
  POKE A,PEEK(1024):POK
  E A+54272,PEEK(55296)
  :WAIT 653,1,1:GOTO 1
```

36

Come per l'una riga 36

(Roberto Morassi)

```
1 PRINT"[RVS] ";:N=N+1+
  40*(N=39):POKE 646,N
  AND 15:GOTO 1
```

37

Scalinatella. Ecco come con poche istruzioni BASIC il vostro Commodore può simulare la rappresentazione di una scala in due dimensioni. Buona discesa!!

(Roberto Morassi)

```
1 A=((N OR 3)/4) AND 1:PRINTMID$("[
  LEFT][DOWN]",A+1,1)CHR$(175-5*A)"
  [RIGHT]";:POKE 646,PEEK(161):N=N+
  1:GOTO 1
```

#38

Serpente. Nel deserto spesso si vedono quei piccoli serpenti che camminano velocemente sulla sabbia. Questa 1R riproduce tale serpente che con la sua strana andatura, quasi sinusoidale, scenderà nella fascia centrale del vostro monitor o TV.

(Roberto Morassi)

```
1 A=20+INT(10*SIN(π*N/13)):PRINT TA
  B(A)"●":N=N+1:POKE 646,PEEK(161):
  GOTO 1
```

#39

Multicolor. L'una riga predispone il Computer per l'uso del carattere multicolore.

(Sandro Mariconda)

```
1 PRINT"[CLEAR]MULTICOL
  OR MODE":B=53280:FOR
  I=1 TO 4:PRINT"COLOR"
  I;:INPUT C:POKE B+1,C
  :NEXT:POKE B-15,91
```

#40

Crea labirinto. Un "intricatissimo" labirinto nel quale non sempre si riesce a trovar l'uscita. Provate!!

(Claudio Levantini)

```
1 PRINTCHR$(205.5+RND(1
  ));:GOTO 1
```

#41

Scritta infinita. Con questa una riga la frase nella variabile A$ verrà scritta all'infinito un carattere alla volta. Ovviamente la frase può essere sostituita con un'altra qualsiasi.

(Ermanno Salvalaio)

```
1 PRINT"[CLEAR]":A$="UNA SOLA RIGA"
  :FOR A=1 TO LEN(A$):PRINT TAB(A)M
  ID$(A$,A,1);:FOR B=0 TO 99:NEXTB,
  A: RUN
```

#42

La colonna vincente. Affidati alla casualità ed utilizza il tuo Commodore per la compilazione della schedina con questa simpatica una riga. (in caso di vincita: il 10% alla redazione)

(Ferdinando Tommasi)

```
1 A$(0)="X":A$(1)="1":A$(2)="2":PRI
  NT"[CLEAR]":FOR I=1 TO 13:A=RND(1
  )*3:PRINTA$(A):NEXT
```

#43

Il topo affamato. Un topolino affamato divora una videata di formaggio svizzero.

(Diego Crema)

```
1 FOR H=1 TO 1000:PRINT
  "[GIALLO][RVS]●";:NEX
  T:PRINT"[HOME][GRIGIO
  2]TOPO";:FOR Y=1 TO 8
  72:PRINT TAB(4)" ";:N
  EXT
```

#44

Diagonale. Ecco una utility carina per abbellire i vostri programmi. Qualunque messaggio tra apici verrà visualizzato in diagonale sullo schermo.

(Marco Benvenuti)

```
1 PRINT"[CLEAR]":A$="SP
  ARARE":FOR I=7 TO 1 S
  TEP -1:PRINTSPC(121)R
  IGHT$(A$,I);:NEXT
```

#45

Riga Colonna. Indovina la riga e la colonna dove è posizionata la chiocciolina ed il computer ti premierà facendoti vedere due segni grafici. Simpatico, no?

(Giovanni Giovanni)

```
1 PRINT"[CLEAR]":A=INT(920*RND(1))+
  1:PCKE 1103+A,0:INPUT X,Y:IF A+12
  0=X+40*Y THEN PRINT"BRAVO"
```

46

Slot Machine. L'una riga si ispira alla famosa macchin e atta da casinò. Diamogli un RUN e la ruota della fortuna comincerà a girare.

(Mario Guarneri)

```
1 FOR A=1 TO 20:PRINT"[CLEAR]"INT(R
  ND(1)*3)+1,INT(RND(1)*3)+1,INT(RN
  D(1)*3)+1:NEXTA
```

47

Gentil Computer. Il tuo computer, Commodore 64, per dimostrarti la sua gratitudine ti saluterà offrendoti la vasta gamma dei suoi colori.

(Giorgio Verdeoliva)

```
1 INPUT "[CLEAR][BIANCO
  ]NOME";A$:B$="CIAO"+A
  $:FOR X=0 TO 255:POKE
   53281,X:FOR B=0 TO 9
  9:PRINTB$:NEXTB,X
```

48

Input al buio. La frase digitata viene visualizzata in un tempo proporzionale alla lunghezza della frase solo dopo aver premuto il tasto RETURN. Tutto ciò per imparare a scrivere al buio.

(Paolo Manarolla)

```
1 INPUT "[CLEAR]DIGITA
  FRASE[BLEU]";A$:PRINT
  "[BIANCO]"A$:FOR I=1
  TO LEN(A$)*150:NEXT:G
  OTO 1
```

49

L'omino allegro. Un simpatico omino che saluta all'infinito chi lo sta ad osservare.

(Fabio Busnelli)

```
1 PRINT"[HOME] ● ":PRINT"/▒\":PRINT
  " | | ":FOR I=1 TO 300:NEXT:PRINT"[
  HOME]\●/":PRINT" ▒ ":PRINT"/ \":F
  OR I=1 TO 300:NEXT:GOTO 1
```

50

Colore. Prima lo sfondo poi il bordo ecco tutti i colori del tuo sistema Commodore.

(Lino Italia)

```
1 FOR I=1 TO 255:POKE 53280,I:FOR K
  =1 TO 50:NEXTK,I:FOR H=1 TO 255:P
  OKE 53281,H:FOR Q=1 TO 50:NEXT:GO
  TO 1
```

51

Lo specchio. Avete presente quei tubi di plastica attraverso i quali da piccoli avvicinando l'occhio vedevamo piccoli frammenti colorati riprodursi magicamente ai lati? Questa routine simulerà l'effetto caleidoscopio sullo schermo del 64 Commodore: basterà inserire delle parole per vederle riprodotte specularmente a lato.

(Roberto Morassi)

```
1 GET A$:PRINT"[BIANCO]
  "A$;:A=PEEK(211):PRIN
  T"[GIALLO]" TAB(39-A)
  A$;:POKE 211,A:GOTO 1
```

52

Terremoto. Una sconvolgente una riga che simula un muovimento sismico. Tenetevi forte e fate girare il programma.

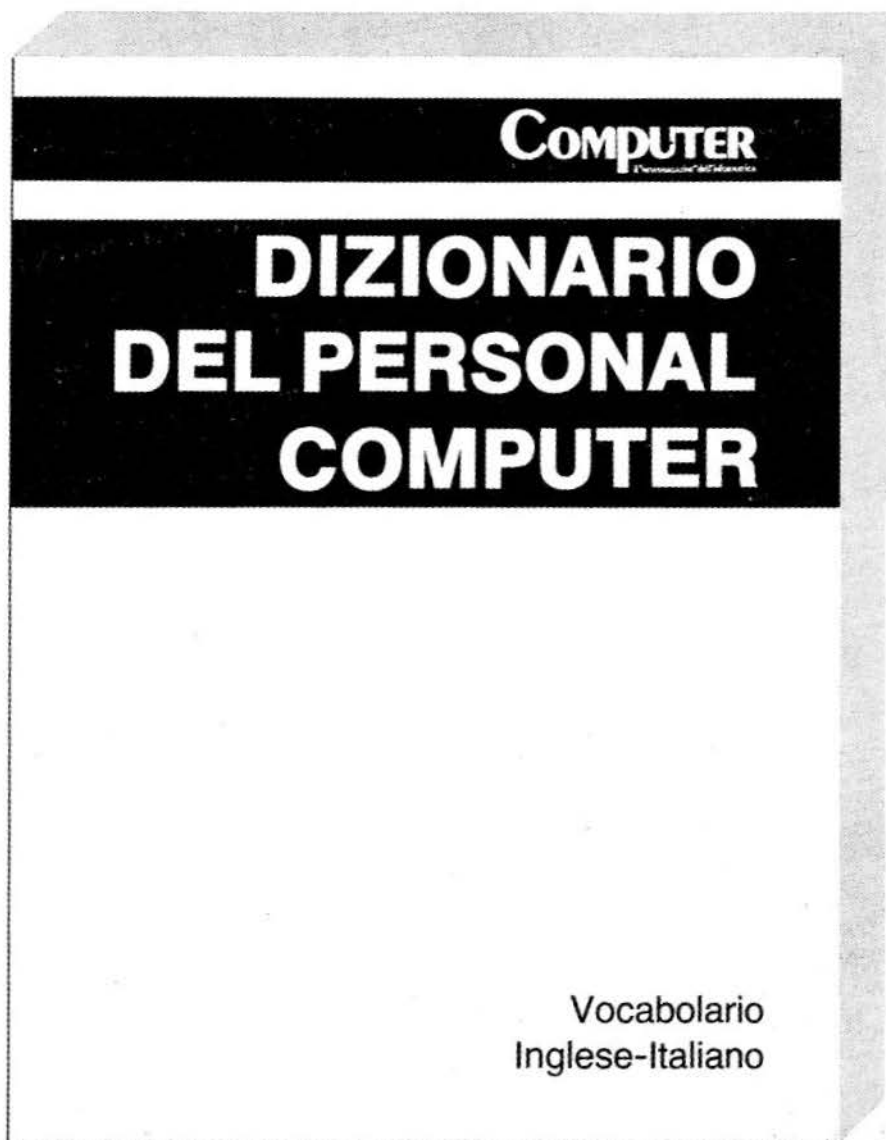
(Antonio Sorrentini)

```
1 FOR I=0 TO 255:PRINTI
  ;:POKE 53270,I:NEXT:G
  OTO 1
```

53

Invertitore. Diamo una parola ed il minuscolo programma la invertirà.

(Andrea Antoniazzi)

```
1 INPUT N$:IF N$>"" THEN A=LEN(N$):
  FOR T=1 TO A:PRINTMID$(N$,A+1-T,1
  );:NEXT:PRINT:GOTO 1
```

COMPUTER

Annuario 1985

N. 73 - lire 7000 — il "newsmagazine" dell'informatica

Hobby & home

Portatili

Trasportabili

Micro & mini

Supermini e mainframe

600 modelli con tutti i dati e i prezzi

Systems

Computer - Anno X - N. 73 - 25 febbraio/25 marzo 1985 - Sped. Abb. Post. Gr. III/70 CR - Distr. MePe

DISK DOCTOR

di **Giancarlo de Cobelli**

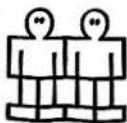
Con la comparsa sul mercato di moltissimi programmi su disco protetti, sono arrivate — in particolare dalla Germania — altrettante utility che permettono di analizzare il disco per scoprire come è stato creato l'errore. Possono essere creati su disco vari tipi di errori, che però si basano tutti sul medesimo concetto.
Il sistema di creazione dell'errrore sta nel forzare determinati dati incoerenti sulle tracce e settori del dischetto. Quando il DOS (Disk Operating System) esamina i contenuti del disco, sia per lettura che per scrittura, esegue anche dei controlli di coerenza su determinati dati che non sono direttamente trasparenti all'utente. Queste forzature, se così vogliamo chiamarle, sono eseguibili con programmi di particolare complessità e varietà.
Il programma che vi propongo non vuole rendere possibile una creazione di errore sul disco, ma esclusivamente permettere un esame del disco stesso con l'individuazione delle possibili incongruenze. Il programma principale contiene una particolare routine allocata in una precisa parte della memoria che controlla, se si verificasse durante la lettura del disco, una particolare condizione su una determinata traccia.
Questa condizione può essere di diversi tipi a seconda dell'errore che viene creato. Di solito si riscontrano i seguenti tipi di errori che sono anche citati sul manuale del floppy drive:

- Errore 21: il controller non è in grado di trovare il sincronismo sulla traccia desiderata.
- Errore 23: nei bytes dei dati registrati può esistere uno o più errori; cioè il dato è stato letto dalla memo-

PROGRAMMA : DISK DOCTOR PRG

VAR.	LINEA DEL PROGRAMMA					
A	127	136	152	153	154	163
	164	165				
A$	136	152	180	200	203	205
	218					
B$	180	200	204			
BY	138	165				
C$	180					
D$	180					
DD$	180					
ID	129	163				
IZ	163	164	165			
MEM	135	163				
MH	135					
ML	135					
NL	138	139				
Q$	128	136				
S	198	199	202	206		
S$	175	176	183	196	197	208
T	161	162	164	166	167	177
	178	179	181	184	198	199
	201	209				
T1	195	198				
T1$	194	195				
T2	195	198				
T2$	194	195				

Tutti i numeri che occorrono per essere il n. 1

delle stampanti in Italia

1. Operare solo nel mercato delle periferiche
2. Produrre più di 350.000 macchine all'anno
3. Offrire una vasta gamma di modelli
4. Essere presente nelle varie fasce di mercato
5. Avere oltre 10 anni di attività in Italia
6. Investire in ricerche tecnologiche per anticipare le esigenze del mercato
7. Garantire una efficace assistenza tecnica su tutto il territorio nazionale

Guarda caso! La

ha tutti i numeri... dal 1 al 7

ria del drive, ma è stato riscontrato un errore di checksum (un valore che serve al DOS per controllare l'esattezza della traccia, ovvero il sincronismo della traccia stessa con le altre presenti sul floppy).

• Errore 27: nell'intestazione di un blocco di dati è stato riscontrato un errore di checksum; cioè il blocco non è stato letto nella memoria del drive.

• Errore 29: viene interpretato come un disco non inizializzato poiché riscontra delle diverse ID nella lettura delle traccie, infatti l'ID viene riportato in una determinata maniera sempre all'inizio di ciascuna traccia.

Per sorpassare questo tipo di problema possiamo seguire due vie: riportare l'errore sul disco copiato o eliminare il controllo dell'errore sul disco originale.

Con i programmi in commercio è molto più semplice riportare l'errore sul disco copiato. Per fare ciò occorre sapere principalmente che errore è stato creato sul disco originale.

Utilizzando il programma proposto è possibile analizzare il disco per sapere se durante la formattazione sono stati creati degli errori.

Nelle prime righe viene inizializzato il drive per evitare qualunque problema in lettura e controllato se il disco non presenta nessun problema.

Le subroutines servono per la lettura della ID e per la relativa conversione in esadecimale.

Il menù presenta quattro opzioni: la prima di queste serve per leggere l'identificatore di ogni singola traccia (questo è molto utile per verificare se esiste l'errore 29, poiché in tal caso gli identificatori sulle traccie non saranno tutti uguali); la seconda opzione permette di analizzare traccia per traccia l'eventuale esistenza degli errori mostrando il classico messaggio della lettura delle variabili, chiamate dal manuale ER, EM$, ET, ES, che ci fanno conoscere il tipo di errore riscontrato, il messaggio, la traccia e il settore dove si è verificato. La terza opzione serve per una analisi più completa, anche se più lunga come tempo, degli errori riscontrati sulle traccie per ogni settore (è possibile esaminare solo le traccie interessante).

La lettura dell'identificatore è basata tutta sul comando Memory Read; questo permette di leggere direttamente buffers che il DOS ha inizializzato sulla RAM scegliendo il byte che si vuole conoscere.

La lettura degli errori utilizza lo stesso principio leggendo i blocchi di bytes senza variare il puntatore del buffer.

A questo punto manca solo un sistema di creazione dell'errore riscontrato sul dischetto originale. Per fare ciò, come ho avuto modo di dire all'inizio è necessario un sistema un po' più sofisticato del semplice programma in oggetto.

In data odierna posso dire che esistono alcuni programmi per ottenere la forzatura di errori, si chiamano Clones e sono ormai di dominio pubblico.

```
100 REM ***************************
101 REM *                         *
102 REM *      DISC  DOCTOR       *
103 REM *                         *
104 REM ***************************
105 REM *                         *
106 REM * AUTHOR SOFTWARE :       *
107 REM *                         *
108 REM *  GIANCARLO DE COBELLI   *
109 REM *                         *
110 REM ***************************
111 REM *                         *
112 REM * VIC 20 + EXP.      NO   *
113 REM * COMMODORE 64       SI   *
114 REM * COMMODORE 4000     NO   *
115 REM * COMMODORE 8000     NO   *
116 REM * COMMODORE 16       SI   *
117 REM * COMMODORE PLUS 4   SI   *
118 REM *                         *
119 REM ***************************
120 :
121 REM ***************************
122 REM *    INIZIALIZZAZIONE     *
123 REM ***************************
124 :
125 POKE 53280,6:POKE 53281,6:PRIN
    T"[CLEAR]"
126 OPEN 15,8,15,"I0"
127 INPUT#15,A:IF A THEN STOP
128 OPEN 2,8,2,"#":Q$=CHR$(0)
129 ID=16+6:GOTO 145
130 :
131 REM ***************************
132 REM *       SUBROUTINES       *
133 REM ***************************
134 :
135 MH=INT(MEM/256):ML=MEM-MH*256:
    PRINT#15,"M-R";CHR$(ML);CHR$(M
    H)
136 GET #15,A$:A=ASC(A$+Q$):RETURN
137 :
138 PRINT"$";:NL=INT(BY/16):GOSUB
    139:NL=BY-16*NL
139 PRINTMID$("0123456789ABCDEF",N
    L+1,1);:RETURN
140 :
141 REM ***************************
142 REM *         MENU'           *
143 REM ***************************
144 :
145 PRINT TAB(15)"[RVS][CLEAR][VIO
    LA]DISC  DOCTOR[RVOFF]"
146 PRINT TAB(12)"[3 DOWN][VERDE]
```

```
   *** MENU ***"
147 PRINT TAB(11)"[3 DOWN][RVS][AR
    ANC] 1.[RVOFF][VERDE] LEGGE ID
    "
148 PRINT TAB(11)"[DOWN][RVS][ARAN
    C] 2.[RVOFF][VERDE] TROVA ERRO
    RI"
149 PRINT TAB(11)"[DOWN][RVS][ARAN
    C] 3.[RVOFF][VERDE] ESAMINA ER
    RORI"
150 PRINT TAB(11)"[DOWN][RVS][ARAN
    C] 4.[RVOFF][VERDE] QUIT"
151 PRINT"[3 DOWN]SCEGLI (1-4) ? "
    ;
152 OPEN 1,0:INPUT#1,A$:CLOSE 1:A=
    VAL(A$):IF A=0 THEN 145
153 IF A>5 THEN 145
154 ON AGOTO 160,175,192,216
155 :
156 REM *************************
157 REM *       LETTURA ID      *
158 REM *************************
159 :
160 PRINT"[CLEAR][DOWN][VIOLA][RVS
    ]LEGGE ID[RVOFF][2 DOWN]"
161 FOR T=1 TO 35
162 PRINT#15,"U1:2,0,";T;",0"
163 MEM=ID:GOSUB 135:IZ=A:MEM=ID+1
    :GOSUB 135:
164 PRINT"[ARANC][RVS]"RIGHT$(STR$
    (T),2);".[RVOFF][VERDE]ID=";CH
    R$(34);CHR$(IZ);CHR$(A);CHR$(3
    4);" ";
165 BY=IZ:GOSUB 138:PRINT",";:BY=A
    :GOSUB 138
166 PRINT TAB(20);:IF T/2=INT(T/2)
     THEN PRINT
167 NEXTT
168 GOSUB 217
169 PRINT"[CLEAR]":GOTO 145
170 :
171 REM *************************
172 REM *     TROVA ERRORI      *
173 REM *************************
174 :
175 S$="[CLEAR][VIOLA][RVS]TROVA E
    RRORI[RVOFF][DOWN]"
176 PRINTS$
177 FOR T=1 TO 35
178 PRINT"[RVS][ARANC]"RIGHT$(STR$
    (T),2);".[RVOFF]";
179 PRINT#15,"U1:2,0,";T;",0"
180 INPUT#15,A$,B$,C$,D$:DD$=A$+",
    "+B$+","+C$+","+D$:PRINT"[VERD
    E] "DD$
181 IF T<>17 THEN 184
182 GOSUB 217
183 PRINTS$
184 NEXTT
185 GOSUB 217
186 PRINT"[CLEAR]":GOTO 145
187 :
188 REM *************************
189 REM *    ESAMINA ERRORI     *
190 REM *************************
191 :
192 PRINT"[CLEAR][VIOLA][RVS]ESAMI
    NA ERRORI[RVOFF][DOWN]"
193 PRINT"[VERDE]CHE TRACCIA (T1,T
    2) ?";
194 OPEN 1,0:INPUT#1,T1$,T2$:CLOSE
     1
195 T1=VAL(T1$):T2=VAL(T2$):IF T1<
    =0 OR T2>35 THEN 194
196 S$="[CLEAR][ARANC][RVS]TRC  SC
    T  ERRORE  MESSAGGIO[RVOFF]"
197 PRINTS$
198 FOR T=T1 TO T2:FOR S=0 TO 99
199 PRINT#15,"U1:2,0,";T;",";S
200 INPUT#15,A$,B$:IF A$="66" THEN
     207
201 PRINT"[VERDE]"RIGHT$(STR$(T),2
    )
202 PRINT"[UP]" TAB(5)RIGHT$(STR$(
    S),2)
203 PRINT"[UP]" TAB(12)A$
204 PRINT"[UP]" TAB(18)B$
205 IF A$="00" THEN PRINT"[UP]"
206 NEXTS
207 GOSUB 217
208 PRINTS$
209 NEXTT
210 PRINT"[CLEAR]":GOTO 145
211 :
212 REM *************************
213 REM *         FINE          *
214 REM *************************
215 :
216 CLOSE 2:CLOSE 15:SYS64738
217 PRINT:PRINT"[VIOLA]PREMI [RVS]
    SPACE[RVOFF] PER CONTINUARE"
218 GET A$:IF A$="" OR A$<>" " THE
    N 218
219 RETURN
```

Scambiatevi le liste

Tutte le richieste di Vendo-Scambio-Compro programmi per... verranno citate esclusivamente con il solo nome, cognome ed indirizzo. Sta a voi lettori scambiarvi le varie liste. Verranno pubblicati integralmente a giudizio del caporedattore, quegli annunci che non fanno parte della categoria di cui sopra. Buoni scambi e vendite!!

(Gloriano Rossi)

Luigi Domuso via Bellani 3 20124 Milano Tel. 02/6705774

Jurgen Morhofer via Lorenzo il Magnifico 148 00162 Roma Tel. 06/4270418

Michele Martino via Fanin 2 71013 S. Giovanni Rotondo Tel. 0882/854289

Claudio Quaglia via Germonio 11 10095 Grudiasco (TO) Tel. 011/702951

Gian Maria Agretti via Del Genio 4/5 40135 Bologna Tel. 051/414630

Alberto Frabetti via Ricciarelli 139 44100 Ferrara Tel. 0532/61034

Giuseppe Borracci via Mameli 15 33100 Udine Tel. 0432/291665

Francesco Cinquini via Barcara 54011 Aulla (MS)

Massimo Andrian via Aquilea 63 35070 Villesse (GO) Tel. 0481/91095

Stefano Brighi via Magliana Nuova 284 00176 Roma Tel. 06/5283428

Andrea Sant via Diaz 18 33010 Treppo Grande (UD) Tel. 0432/960604

Simone Lombardi via V. Emanuele 205 55041 Camaiore (LU) Tel. 0584/989728

Roberto Cavataio via G. Agnelli 1 10070 Robassomero (TO) Tel. 011/9295086

Giuseppe Bellinghieri Cas. Postale 53 Contesse 98100 Messina Tel. 090/2712633.

Paolo Rivieccio via Montebello 115 57100 Livorno Tel. 0586/807043

Francesco Basile via Ortigara 14 31100 Treviso Tel. 0422/579627

Maurizio Parenzan via Gabrieli 13 70125 Bari Tel. 080/368291

Marco Pendino via Betti 25 20100 Milano Tel. 02/3087174

Stefano Barcaroli via B. Buozzi 8 05021 Acquasparta (TR) Tel. 0744/930387

Massimo Mazzoli via E. Fermi 5 42022 Boretto (RE) Tel. 0522/686155

Francesco Gin via S. Croce 1315 30125 Venezia Tel. 0421/81241

Guido Demichelis Guido via Sondrio 9 10144 Torino

Silvano Bompieri via Baccaglioni 8 46040 Monzambano (MN) Tel. 0376/845372

Alessandro Zanellato Condominio Ciclamino 10090 Romano Canavese (TO)

Luca Viola via Priaruggia 26/1 16030 Pieve Ligure (GE) Tel. 010/3460475

Michele Petracca via Donatello 12 35027 N. Padovana Tel. 049/627164

Roberto Manfroni via B. Buozzi 4 00049 velletri (ROMA) Tel. 06/9636898

Fabrizio Rizzi via Castello 3060B 30122 Venezia Tel. 041/22883

Carlo Pezza via S. di Santarosa 61 00149 Roma Tel. 06/5281016

Sandro Natali via XVI Luglio 36 50019 Sesto F.no (FI) Tel. 055/448140

Andrea Buongiorni via Cipeli 42 28100 Piacenza Tel. 0523/73071

Marco Schito via Lucca 36 20152 Milano Tel. 02/4591526

Giuseppe Di Lello C.so Europa 13 66054 Vasto (CH) Tel. 0873/4393

Claudio Giacchetta via Cave 36/A 60020 Sirolo (AN) Tel. 936098-936196

Cristiano Stasi via Smaldone 21 71100 Foggia Tel. 0881/86086

Istituto Educativo per scambi senza fine di lucro, dispone di oltre 200 programmi per VIC-20 (giochi didattica) e di oltre 1000 programmi per Commodore 64 (giochi, didattica, linguaggi, utility, package). Preferibilmente zona Napoli, per contatto diretto Tel. 7418802 ore 10-16.

Istituto "Opera per la salute del fanciullo" via Bellaria 4 Napoli è interessato a visionare e valutare eventuale collaborazione per sofware di tipo didattico per la fascia della scuola materna ed elementare implementabile su Commodore 64 e VIC—20. Tel. 081/7418802 ore 10-16.

Cerco espansioni RAM e cartridges (es. Forth, Wordcraft, Programmer Aid's) per VIC-20. Offro in cambio programmi a scelta, e nella quantità desiderata tra oltre 200 titoli (i migliori).
Vincenzo Carrone via G. Pascoli 67 86100 Campobasso Tel.0874/91995

Vendo Commodore VIC-20 più registratore. Raffaele Lubrano via Virglio 62 20017 Rho Tel. 02/93169656

Vendo Commodore 64 più copritastiera, registratore 1530, più vari giochi ed utilities. Il tutto usato pochissimo a lire 600.000.
Rolando Carlini via Stoccolma 51 47037 Rimini Tel. 0541/33868

Vendo Commodore VIC-20 usato solo

2 mesi al fantastico prezzo di lire 159.000,più in regalo un joystick una cartuccia e 5 cassette (prezzo trattabile).
Mauro Imbriani via Veglie 83 73041 Carmiano (LE) Tel. 0832/671562.

Vendo stampante Commodore MPS-802 nuovissima con garanzia e fattura, in regalo un Word Processor (Hes Writer), a lire 450.000.
Michele Reboli via Bertieri 7 20146 Milano Tel.02/471541.

Vendo interfaccia che collega il Commodore 64 con qualsiasi tipo di registratore a lire 25.000 comprese spese postali.
Massimo Mazzoli via E. Fermi 5 42022 buretto (RE) Tel. 0522/686165

Compro Commodore 64 a prezzo ragionevole. Oppure cambio con Atari VCS 2600 corredato di 7 cartucce più Commodore VIC—20 espanso a 16K con 100 programmi su cassetta.
Giuseppe Paterniti via Nazionale Cond. Etna 28030 Trappitello (ME)

Vendo dischetti Nashua a lire 3700 e programmi per Apple IBM e Commodore 64.
Luigi Domuso via Bellani 3 20124 Milano Tel. 02/6705774

Vendo VIC-20 più 16K più cartuccia Hesmon più registratore originale con le migliori utility ed i migliori giochi. Il tutto a lire 350.000.
Riccardo De Mattia via Rinuccini 3 20125 Milano Tel. 02/2870233.

Vendo per passaggio a sistema superiore, VIC 20 con registratore C2N più espansione grafica da 3 KRAM + joystick + penna ottica + moltissimo software commerciale su cassetta. Il tutto corredato da manuali e perfettamente funzionante a lire 400.000 (Toschi Vladimiro - Via Bentivogli 32 - 40055 Castenaso (BO) - 051/784271 dopo le ore 12).

Vendo VIC 20 + registratore C2N + joystick + Basic parte 1 + 80 giochi in cassetta + 1 cartridge + il libro "Alla scoperta del Vic 20" a lire 300.000 non trattabili. (Claudio Spadazzi - Via Tito Papiro 14 - 47100 Forlì - 0543/35602 ore pasti).

Vendo computer CBM-PET-3032 nuove ZROM amplificatore radio incorporato manuali d'uso Basic, aggiornamenti, libri programmi: W.P., EXT. Basic, assembler, dos support, scacchi 8 livelli L.M. e decine di giochi il tutto a lire 1.000.000. (Basso Gianfranco - Via Parmitta 24 - 35100 Padova - 049/756252 orario pasti).

Vendo Vic 20 nuovissimo settembre '84 usato pochissimo + 2 manuali + 4 cassette gioco tutto a lire 200.000 (Claudio Ciavatta - Via Maroncelli 2 - 47036 Riccione - 0541/604666 orario pasti).

Vendo per Vic 20 nuovissima scheda che rende la registrazione dei programmi veloce quanto il disk drive (25K al minuto), con connettore per eventuali espansioni lire 35.000. Vendo inoltre scheda Rom che oltre a portare a 40 colonne il Vic lo rende compatibile PET/CBM a lire 35.000. (Gianni Bozzi - Via Savona 16/s - 20099 S.S. Giovanni (MI) - 02/2407825).

Compro copia dei numeri 5 e 6 di Commodore Computer Club. (Franco Spada - Via Fardella 444 - 91100 Trapani - 0923/22301 ore ufficio).

Vendo traduzione delle istruzioni del videogioco "Solo Flight", simulazione di volo, lire 5000. (Budal Anna - Località Log 192 - 34010 S. Dorligo Valle (TR) 040/813145).

Vendo "Basic enciclopedia dell'informatica dei mini e personal Computer" completa di copertine e frontespizzi aggiornata all'ultimo fascicolo uscito (o eventualmente terminata). Mai sfogliata per mancanza di tempo con sconto del 20% del prezzo di copertina. (Giorgio Tonem - Via Comano 35 - 54036 Marina di Carrara - 0585/634283 dopo le 19).

Vendo Extended Basic per CBM-64 completo di istruzioni a lire 40.000 (Pietro Nieddu - Via Bianchi 1 - 07100 Sassari)

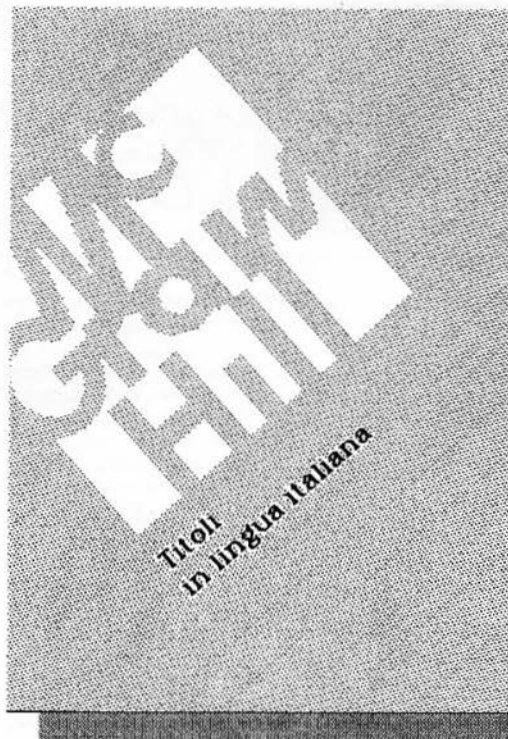
Vendo per Vic 20 espansioni 3K super Grafix o 16K rispettivamente a lire 50.000 e 110.000 + lire 2000 per spese spedizioni. Libro di grafica con 38 programmi per 3K super Grafix a lire 10.000. Libro "Alla scoperta del Vic 20" a lire 15.000 (Ferrari Fabio - Via Araldi 5 - 29100 Piacenza - 0523/753283 dopo le ore 18).

Vendo o cambio 2 cartucce per CBM-64, Avanger e Kick Man con Music Machine + Music Composer. (Porcellana Claudio - Via Olmi 5 - 10048 Vinvo (TO) - 9651559 ore pasti).

Vendo Vic 20 (causa passaggio a sistema superiore) + registratore dedicato + joystick professionale + 3 manuali + 1 cassetta con 40 stupendi giochi + oltre riviste e più di 300 programmi. Il tutto in ottime condizioni a lire 380.000 trattabili anche in pezzi separati. (Giovanni Azzara - Via Peone 3 - 20089 Rozzano (I) - 02/8240807 pre 20 alle 22).

Vendo Commodore Vic 20 + cartridge "Hes writer" + cartridge "Super expander" + circa 200 Video Games per il modico prezzo di lire 230.000 (anche singoli pezzi). Tutto perfettamente funzionante. (Michele Pozzi - Via Coll. M. Luigia 15 - 43100 Parma - 0521/32165 ore pasti).

Vendo causa passaggio a CBM-64, Vic 20 + 70 giochi su cassetta a lire 200.000 (non trattabili). Il sistema è in ottimo stato ed efficentissimo (Monticelli Luca - Via F.lli Bressan 4 - 20126 Milano - 02/2551762 ore pasti).

Vendo stampante GP-100-VC nuova, collegabile direttamente al CBM-64, a lire 350.000 non trattabili, preferibilmente zona Milano e provincia. (Parnigoni Fulvio - Via Foscolo 41 - 20050 Lesmo (MI) - 039/6980089 ore dalle 12 alle 13).

IN LIBRERIA

J.Heilborn, R.Talbott
Guida al Commodore 64
pag.440 L.36.000
...tre ristampe in 10 mesi!

R.Jeffries, G.Fisher e
B.Sawyer
Divertirsi giocando con il Commodore 64
pag.280 L.22.000

H.Peckham
Il BASIC e il Commodore 64 in pratica
pag.312 L.27.000

marzo '85
K.Skier
L'Assembler per il Commodore 64 e il VIC 20
pag.400 L.35.000

SOFTWARE su cassetta
Commodore 64 Assembler / Disassembler (in preparazione)

La McGraw-Hill pubblica in tutto il mondo decine di titoli dedicati ai calcolatori della Commodore.
Richiedete il catalogo dei libri in lingua italiana e il McGraw-Hill Computer Catalogue per la produzione in lingua inglese.

distribuzione in libreria:
Messaggerie Libri S.p.A.
Via Giulio Carcano 32
20141 Milano

McGraw-Hill Book Co. GmbH
Lademannbogen 136
D 2000 Hamburg 63
Rep. Federale Tedesca

ELETTRONICA VALDARNESE s.d.f. Via Marconi 9/A-Loc.Muraccio
52025 MONTEVARCHI (AR) tel. 055/980242-982513 C/c postale N.10418523

ESTRATTO DEL CATALOGO GENERALE

PER COMMODORE 64

GESTIONALI

CONT.GENERALE(D)	180.000
Fatturazione(D)	120.000
Magazzino(D)	120.000
Gestione negozi(D)	150.000
Cartella clinica(D)	150.000
Mailing list(D)	60.000
Agenda telef.(D)	60.000
ARREDOGRAPH	195.000

WORD/PROC.

Easy script(D/N)	70.000
Vizawrite (D)	75.000
Word Pro.III (D)	75.000

UTILITY/VARIE

Compilatore DTL(N)*	40.000
Austro Compiler(D)	70.000
Pet Speed (D)	70.000
PASCAL OXFORD(D)*	150.000
Assembler (N)	35.000
" (D)*	60.000
" (C)	70.000
KMMM Pascal (D)	80.000
Supermoon (N)	30.000
" (C)	60.000
Turbo tape (N)	28.000
Fast copy (D)	50.000
Unguard (D)	120.000
The Clone (D)	80.000
Disk Doctor (D)	50.000
80 colonne (D/N)	40.000
64 Diagnosys (D/N)	40.000
Master (D)	110.000
Tool (D)	70.000
The Manager (D)*	120.000
Calc Result Easy(C)	95.000
" Exp(C+D)	160.000
Extended basic (C)	75.000
Compactor (D)	30.000
Scompactor (D)	30.000
Superbase (D)	120.000
Basic Wedge (C)	95.000
Toto 13 (D/N)	60.000
TURBO DISK (D)	60.000
FAST FORMATTER	40.000
ISAM 64	75.000
Character Editor	28.000
Sprite Editor	28.000
Protector	250.000
Chiavi protezione	50.000

GRAFICA/MUSICA

Ultrabasic (D)	70.000
Pictograph (C)	69.000
Magic paint (D)	70.000
Koala paint (D)	95.000
Panorama (D)*	65.000
Synthy (D)*	70.000
SAM RECITER (D)	80.000
MUSICALC 1/2/3 (D)	120.000

PER VIC 20

Mailing list(D)	60.000
Magazzino (D)	95.000
Fatturazione(D)	95.000
Monitor (N)	28.000
40 colonne (N)	22.000

H A R D W A R E

Cavo centronics	38.000
Int. 64-Centronics	95.000
Pet/IEEE-Centr.	120.000
Buffer 8K Centr.	220.000
Int. ET 121-221	250.000
Monitor Verde 12"	179.000
" Arancio	189.000
Stamp. Tally MT80	690.000
Espan. 16K VIC	118.000
" 32K VIC	145.000
Joystick	22.000
Dischi scat. 10	39.500
Nastri C10-20-30x10	12.000
Vic Eprom progr.	180.000
Vic Mot.Bo.(4slots)	59.000

G I O C H I

RICHIEDERE CATALOGO PARTICOLARE

MANUALI IN ITALIANO

Pet Speed	15.000
Easy Script	20.000
Simon Basic	20.000
Master	25.000
Tool	15.000
Superbase	25.000
Vizawrite	15.000
Colossus (scacchi)	3.000
The Clone	10.000
Unguard	10.000
Statistica	12.000
Multiplan (HELP)	12.000
KMMM Pascal	10.000
Pictograf	5.000
Word Pro.III	7.000

GUIDA AL CBM 64 25.000
**Nuova edizione,riveduta e amplifata. L'unico con il Commodore Approved.

SISTEMA OPERATIVO 64 38.000
**Questa edizione viene fornita con un programma Disassembler, Assembler, Monitor (N)

I SEGRETI DEL 1541 28.000

Tutto cio' che e' necessario sapere sul disco.Sistema Operativo disassemblato, la Pagina Zero, le routines. I Relatives approfonditi. Il SORT, gli OVERLAY. Tecniche di protezione e sprotezione. Monitor per disco. Disassemblatore DOS e disco.

PERIFERICHE COMMODORE 25.000

**Questo manuale e' stato scritto per insegnare a comprendere ed usare TUTTE le periferiche dei prodotti COMMODORE. 430 pagine. Files relatives su 1541. Lettura e scrittura dati e funzionamento HARDWARE. Tavole BAM e DIRECTORY. Utilizzo delle porte IEEE-488,IEEE seriale, RS-232. Spiegazioni di quasi tutte le stampanti: 1515, 1525, 1526, MPS801, MPS802, 3022, 4022. Numerosi programmi fra cui: RECUPERO FILES, CROSS REFERENCE e addirittura un DATA BASE.

GUIDA AL PERSONAL VIC/20 25.000
**Il piu' completo manuale che vi SVELA come e' costruito e come funziona questo computer. Collegamenti elettrici, mappe di memoria, il linguaggio macchina. Tutta la grafica gestibile ed il suono.

CORSO DI GRAFICA 24.000

Come utilizzare la grafica anche senza essere programmatori. Come scrivere i giochi e come dare il movimento alle immagini. 12 lezioni, 4 programmi di base oltre 40 pagine di tavole.

ACCOPPIATORE ACUSTICO

E' la grande novita' che permette di collegarsi a qualsiasi banca dati, scambiarsi programmi e notizie fra utenti. Disponibile prestissimo un centro di collegamento e scambio presso EVM. CHIEDERE OPUSCOLO GRATUITO sull' accoppiatore, modem e banche dati.

Nome ________________

Cognome ________________

Via ________________

C.A.P. ____ Citta' ________________

INVIATEMI:

O CATALOGO

O CATALOGO GIOCHI

O ________________

LEGENDA/CONDIZIONI

D=Disco / N=Nastro / C=Cartridge / *=Con manuale in Inglese

I prezzi,tranne che per i manuali,sono al netto di IVA.Per spedizioni in contrassegno,calcolare £. 5.500 per spese postali e varie. Con pagamento anticipato SPEDIZIONE GRATUITA.SCONTI PER I SIG. RIVENDITORI. CATALOGO GRATUITO A RICHIESTA. INSERIMENTO GRATUITO IN LISTA DI AGGIORNAMENTO. TUTTI I MESI NOVITA' SENZA IMPEGNO.

Se vuoi abbonarti

Registrate il mio abbonamento annuale a Commodore.

☐ Ho versato oggi stesso il canone di Lire 28.000 a mezzo c/c postale n° 31532203 intestato a: Commodore Systems Editoriale - V.le Famagosta, 75 - 20142 Milano

☐ Accludo assegno per lire 28.000 banca .... n° .... a favore di Commodore Systems Editoriale

Il mio computer è: VIC 20 ☐, C 64 ☐, altro (specificare) ....

Ho ☐ / non ho ☐ la stampante, ma voglio ☐ comprarla.

Preferisco programmi di gioco ☐, didattici ☐, d'utilità ☐, altro ....

Nome .... Cognome ....

Via .... n° .... CAP. .... Città ....

Tel. ....

Se vuoi collaborare

Registratemi fra i collaboratori regolari di Commodore.

A titolo di prova vi invio un articolo e la cassetta col programma "........" di cui vi garantisco l'assoluta originalità autorizzandovene la pubblicazione.

☐ Scrivetemi all'indirizzo sottoindicato ....

Nome ....

Via .... N° ....

Tel. .... CAP .... Città ....

Se vuoi un consiglio o consigliarci

HELP ....

Nome ....

Via .... n° .... CAP. .... Città ....

Tel. .... Orario ....

Il mio computer è configurato:

			Sono in possesso	No	Ho intenzione di acquistare
Vic 20 ☐ C 16 ☐ Plus 4 ☐ C 64 ☐			☐	☐	☐
			☐	☐	☐
Floppy ☐	quale: 1541 ☐	altro	☐	☐	☐
Stampante ☐	quale: MPS801 ☐	altro	☐	☐	☐
Plotter ☐	quale: 1520 ☐	altro	☐	☐	☐
Registratore ☐	quale: 1530 ☐	altro	☐	☐	☐

Televisore ☐, TV-Monitor ☐, Monitor ☐, Colore ☐, B/N ☐

Nome .... Cognome ....

Via .... n° .... CAP. .... Città ....

Tel. ....

Se vuoi vendere o comprare

Vendo ☐ Compro ☐ ....

Nome ....

Via .... n° .... CAP. .... Città ....

Tel. .... Orario ....

Da inviare in busta chiusa a:

Spett.le rivista
Commodore
Systems Editoriale

v.le Famagosta, 75
20142 Milano

Si, voglio abbonarmi

Da inviare in busta chiusa a:

Spett.le rivista
Commodore
Systems Editoriale

v.le Famagosta, 75
20142 Milano

Si, voglio collaborare

Da inviare in busta chiusa a:

Spett.le rivista
Commodore
Systems Editoriale

v.le Famagosta, 75
20142 Milano

Si, chiedo consiglio

Da inviare in busta chiusa a:

Spett.le rivista
Commodore
Systems Editoriale

v.le Famagosta, 75
20142 Milano

Si, voglio votare

Da inviare in busta chiusa a:

Spett.le rivista
Commodore
Systems Editoriale

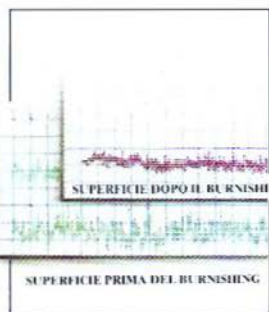
v.le Famagosta, 75
20142 Milano

Si vendo/ compro

SCOPRI LA DIFFERENZA DYSAN

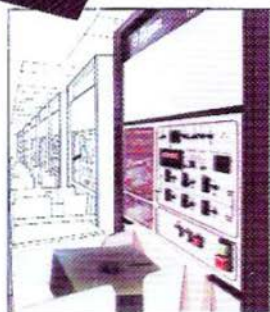
Perchè *Dysan*? Le Quattro Ragioni Per Preferire la Differenza Dysan

1. 100% di superficie testata "error free"

Solo Dysan garantisce che tutta la superficie della diskette sia realmente 100% "error free": un test esclusivo certifica le tracce e lo spazio tra le tracce assicurando prestazioni "error free" anche in presenza di disallineamento delle testine.

2. Esclusiva tecnica di Burnishing

Solo Dysan garantisce una superficie "a specchio" grazie alla sua avanzata ed unica tecnica di "burnishing" - questo risultato assicura un miglior segnale sulle tracce, una minor turbolenza sulle testine, consentendo un sicuro mantenimento dei dati dopo milioni e milioni di rotazioni.

3. Speciale lubrificazione

Solo Dysan garantisce, mediante uno speciale procedimento di lubrificazione, ottenuto trattando la superficie con il proprio esclusivo lubrificante DY 10, che le prestazioni "error free" siano esaltate e mantenute nel tempo.

4. Certificazione totale

Solo Dysan garantisce, con il suo metodo automatico di controllo qualità di tutta la produzione (risultato di una tecnologia leader nel mondo) che ogni diskette prodotta sia stata singolarmente testata e certificata.

Datamatic S.p.A.
via Volturno, 46
20124 Milano
tel.: 02/6073876 (5 linee r. a.)
telex: 315377 SADATA I

Filiale di Roma
via Città di Cascia, 29
tel. 06/3279987